

Domácí úkol 4

Termín odevzdání:	05.04.2020 23:59:59
Hodnocení:	5.0000
Max. hodnocení:	5.0000 (bez bonusů)
Odevzdaná řešení:	1 / 20
Nápovědy:	0 / 10

V tomto úkolu implementujeme stav hry, reprezentace tahů a akce jednotek.

Pravidla hry, které budete v průběhu jednotlivých úkolů implementovat, najdete **zde**.

Zadání úkolu

1. Pokračujte v Java projektu, který jste vytvořili v předchozím úkolu.
2. Z přípravy (stáhněte si níže "Vzorová data") si vezměte šablonu třídy GameState a GameResult. Instance třídy GameState zabaluje celkový stav hry, tedy armády obou hráčů a výsledek hry, tedy jestli se ještě hraje nebo už někdo vyhrál, nebo nastala remíza. V této šabloně implementujete následující metody.

```
// Vrátil dlaždici, která se nachází na hrací desce na pozici pos.
// Musí tedy zkontrolovat, jestli na této pozici není jednotka z
// armády nějakého hráče a pokud ne, vrátí dlaždici z objektu board
public Tile tileAt(TilePos pos)
```

```
// Vraťí true, pokud je možné ze zadané pozice začít tah nějakou
// jednotkou. Vrací false, pokud stav hry není IN_PLAY, pokud
// na dané pozici nestojí žádná jednotka nebo pokud na pozici
// stojí jednotka hráče, který zrovna není na tahu.
// Při implementaci veme v úvahu zahájení hry. Dokud nejsou
// postaveny stráže, žádné pohyby jednotek po desce nejsou možné.
private boolean canStepFrom(TilePos origin)
```

```
// Vraťí true, pokud je možné na zadanou pozici dokončit tah nějakou
// jednotkou. Vrací false, pokud stav hry není IN_PLAY nebo pokud
// na zadanou dlaždici nelze vstoupit (metoda Tile.canStepOn).
private boolean canStepTo(TilePos target)
```

```
// Vráti true, pokud je možné na zadané pozici vyhodit soupeřovu jednotku.
// Vrací false, pokud stav hry není IN_PLAY nebo pokud
// na zadané pozici nestojí jednotka hráče, který zrovna není na tahu.
private boolean canCaptureOn(TilePos target)
```

```
// Vráťí true, pokud je možné na zadanou pozici položit jednotku ze
// zásobníku.. Vrací false, pokud stav hry není IN_PLAY, pokud je zásobník
// armády, která je zrovna na tahu prázdný, pokud není možné na danou
// dlaždici vstoupit. Při implementaci vezměte v úvahu zahájení hry, kdy
// se vkládání jednotek řídí jinými pravidly než ve střední hře.
public boolean canPlaceFromStack(TilePos target) {
```

3. Ve třídě `GameState` si prohlédněte implementaci metod `stepOnly`, `stepAndCapture`, `captureOnly` a `placeFromStack`. Tyto metody zajišťují provádění všech tahů ve hře.
4. Z přípravy si vezměte abstraktní třídu `Move`, která představuje jeden tah v naší hře. Každý tah má clovou pozici `target` a metodu `execute()`, která ze stavu hry `originState` vyrobí nový stav hry. Všimněte si implementovaných metod `hashCode()` a `equals()`. Tato třída má potomky: `StepOnly`, `CaptureOnly`, `StepAndCapture` a `PlaceFromStack`, kteří představují všechny tahy, které lze ve hře provést. Tyto třídy pouze volají příslušné metody v třídě `GameState`.
5. Z přípravy si vezměte abstraktní třídu `TroopAction`, která představuje akci jednotky (krok, posun, úder apod.). Každá instance této třídy obsahuje `offset`, který říká, kde se akce nachází vzhledem k pivotu jednotky. Všimněte si metody `movesFrom`. Ta obdrží pozici jednotky na hrací ploše, stranu za kterou jednotka hraje a stav hry. Na základě těchto údajů vygeneruje všechny možné tahy, které může jednotka pomocí této akce provést v zadaném stavu hry.
6. Pro inspiraci si prohlédněte implementaci třídy `ShiftAction`. Implementuje podle pravidel hry ostatní dvě akce `StrikeAction` a `SlideAction`.
7. Přidejte do třídy `Troop` dva seznamy: `aversActions` a `reversActions`. Tyto seznamy budou obsahovat akce pro rubovou a lícovou stranu jednotky. Upravte všechny tři konstruktory této třídy aby jako parametry očekávaly také tyto dva seznamy. Dále v této třídě implementujte následující metodu:

```
//Vrací seznam akcí pro zadanou stranu jednotky
public List<TroopAction> actions(TroopFace face)
```

8. Z přípravy si vezměte upravené rozhraní `Tile`. Přidává novou metodu `movesFrom`, která vrátí všechny tahy, které lze z této dlaždice provést. V balíčku také najdete upravenou implementaci rozhraní `BoardTile`, které tuto metodu implementuje. Upravte vaši třídu `TroopTile` tak, aby implementovala metodu `movesFrom` tak, že projde všechny akce na správné straně jednotky a sjednotí jimi vrácené tahy do jednoho seznamu.
9. Z přípravy si vezměte třídu `StandardDrakeSetup`, která vytváří všechny jednotky ve hře a využívá nové konstruktory třídy `Troop`. Pomocí ní již můžete vytvořit stav hry a zkusit odehrát několik tahů.
10. Ověřte, že váš kód splňuje testy (stáhněte si níže "Vzorová data"). Spuštějte vždy třídu `MainSuite.java`.

Odevzdání úkolu

1. Vytvořte soubor `thedrake.zip` s přesně touto strukturou složek a souborů

```

/thedrake/
/thedrake/Army.java
/thedrake/Board.java
/thedrake/BoardMove.java
/thedrake/BoardPos.java
/thedrake/BoardTile.java
/thedrake/BoardTroops.java
/thedrake/CaptureOnly.java
/thedrake/GameResult.java
/thedrake/GameState.java
/thedrake/Move.java
/thedrake/Offset2D.java
/thedrake/PlaceFromStack.java
/thedrake/PlayingSide.java
/thedrake/PositionFactory.java
/thedrake/ShiftAction.java
/thedrake/SlideAction.java
/thedrake/StandardDrakeSetup.java
/thedrake/StepAndCapture.java
/thedrake/StepOnly.java
/thedrake/StrikeAction.java
/thedrake/Tile.java
/thedrake/TilePos.java
/thedrake/Troop.java
/thedrake/TroopAction.java
/thedrake/TroopFace.java
/thedrake/TroopTile.java

```

2. Než budete úkol posílat, seznamte se s pravidly pro odevzdávání úkolů (viz. Moodle). Najdete zde důležité informace nezbytné k tomu, aby byl váš úkol přijat.

Vzorová data:

Download

1 17.03.2020 21:27:35

Download

Stav odevzdání:	Ohodnoceno
Hodnocení:	5.0000

- **Hodnotitel: automat**
 - Program zkompilován
 - Test 'Standard test': Úspěch
 - Dosaženo: 100.00 %, požadováno: 100.00 %
 - Úspěch v závazném testu, hodnocení: 100.00 %
 - Celkové hodnocení: 100.00 %
- Celkové procentní hodnocení: 100.00 %
- Celkem bodů: $1.00 * 5.00 = 5.00$