

Mario Kötelező Progrma

A feladat egy Super-Mario szerű játékot játszó ágens implementálása.

A játék során Mariot kell átnavigálni a pályán, vagy minnél meszebb eljutni vele, úgy hogy közben a lehető legtöbb pontot gyűjtsük össze.

Szabályok

Mario a játék bal felén indul és el kell jutni a pálya jobb széléig. Közben lehetőség van érmék felvételére, valamint ajándékok megszerzésére. Továbbá a pálya alsó részén lévő lyukakat lehetőleg el kell kerülni.

A játék véget ér, ha

- elérjük a pálya jobb oldalát
- beleesünk egy lyukba a pálya alján
- 1000 iteráció (interakció) végrehajtása után

Pontozás

A pályán minnél távolabb haladunk a cél fele, annál több pontot szerzünk, pontosabban pálya-egységenként (cellánként) 10-el növekszik a pontszám. Egy érme felvétele (keresztül menni rajta) 100 pontot, egy ajándék megszerzése (alulról ugorva kiütni) 500 pontot ér.

Célok és Paraméterek

- Cél: elérni a pálya jobb szélét, közben minnél több pontot gyűjteni
- Pálya mérete: 13x100, azaz 13 sor és 100 oszlop
- Lehetséges akciók, Mario irányítása:
 - `MarioGame.RIGHT`: jobbra mozgá
 - `MarioGame.UP`: uggrás
 - `MarioGame.LEFT`: balra mozgás
 - `null`: ne csináljon semmit
- A pálya elemei:
 - `MarioGame.EMPTY`: üres (space)
 - `MarioGame.WALL`: fal (#)
 - `MarioGame.PIPE`: cső (P)
 - `MarioGame.COIN`: érme (C)
 - `MarioGame.SURPRISE`: meglepetés (?)
 - `MarioGame.MARIO`: Mario (M)

Keretrendszer

A megoldást Java nyelven kell megvalósítani, egy általunk definiált absztrakt osztály megvalósítása által (részletek később). Az ehhez szükséges keretrendszer a coospace felületről letölthető, használatát pedig a továbbiakban részletezzük.

Szükséges a Java sdk 8 vagy újabb telepítése a fordításhoz és a kiértékeléshez.

Játék indítása vizualizációs felülettel

- Véletlenszerű irányt választó ágenssel: `java -jar game_engine.jar 60 game.mario.MarioGame 1234567890 1000 game.mario.players.RandomPlayer`
- Jobbra-fel mozgó ágens (véletlenszerűen választja a jobbra vagy az ugrás irányokat): `java -jar game_engine.jar 60 game.mario.MarioGame 1234567890 1000 game.mario.players.RightUpPlayer`

Paraméterek:

- 60: debug paraméter (x frames/sec, 0: nincs gui, -x csak konzol kimenet)
- `game.mario.MarioGame`: játék logikát megvalósító osztály
- 1234567890: random seed
- 1000: rendelkezésre álló összigidő (millisec)
- `game.mario.players.RandomPlayer`: Mario-t vezérlő osztály

Saját ágens készítése:

- Hozzuk létre egy `SamplePlayer.java` állományt, a következő tartalommal:

```
import java.util.Random;

import game.mario.Direction;
import game.mario.MarioGame;
import game.mario.MarioPlayer;
import game.mario.utils.MarioState;

public class SamplePlayer extends MarioPlayer {

    public SamplePlayer(int color, Random random, MarioState state) {
        super(color, random, state);
    }

    @Override
    public Direction getDirection(long remainingTime) {
        Direction action = new Direction(MarioGame.DIRECTIONS[random.nextInt(MarioGame.DIRECTIONS.length)]);
        state.apply(action);
        return action;
    }
}
```

- Fordítsuk le a file-t: `javac -cp game_engine.jar SamplePlayer.java`
- Értékeljük ki: `java -jar game_engine.jar 0 game.mario.MarioGame 1234567890 1000 SamplePlayer`
- Kimenet az output csatornán:

logfile: gameplay_xxxxxxx.data
0 SamplePlayer 125.99999999999986 970442441

- Egy játék visszánézése a logfile alapján (25fps): `java -jar game_engine.jar 25 gameplay_xxxxxxx.data`

A játék kimenete:

- 0: játékos azonosító
- `SamplePlayer`: játékos implementáló osztály neve
- 125.99999999999986: elért pontszám
- 970442441: megmaradt gondolkodási idő nanomásodpercben

Kiértékelés

A feladat beadása a coospace-en keresztül történik majd, a beadáshoz egyetlen java file feltöltése szükséges ami a fentiek szerint a stratégia megvalósítását tartalmazza. A keretrendszer használ véletlen döntéseket, tehát a random seed a saját megvalósítás esetleges véletlen döntéseit befolyásolja és attól függetlenül a keretrendszerét is.

Korlátok, határidők, követelmények

- Maximális gondolkodási idő: 1000 ms
- Maximálisan felhasználható memória: 2G
- A teljesítéshez legalább 2000 pontot kell elérni a 10 játékból legalább 8 esetben
- 10 próbálkozás áll rendelkezésre
- Beküldési határidő: 2025. december 1. 23:59

A fenti korlátoknak megfelelő futtatási paraméterezés lehet a következő: `java -Xmx2G -jar game_engine.jar 0 game.mario.MarioGame 1234567890 1000 SamplePlayer`

A kiértékelés során 10 véletlen inicializáció lesz használva (random seed).

További követelmények a megoldással szemben

A megoldásnak saját munkának kell lennie. Konzultáció, közös ötletelés megengedett, de a megvalósítás önálló kell legyen. A megoldást tartalmazó forráskódnak minden körülmények között ki kell elégítenie a következő követelményeket:

- A megoldás nem állhat előre legyártott lépéssorozat visszajátszásából
- A forráskódot `Agent.java` néven kell feltölteni
- A feltöltött forráskódnak le kell fordulnia és hibamentesen le kell futnia

- A feltöltött fájlt az `iconv -f ascii -c` paranccsal ASCII-vé konvertáljuk a fordítás előtt. Emiatt az ékezetes betűk és minden más nem-ascii karakter ki lesznek vágva, tehát jobb ezeket eleve kerülni. Javasolt az UTF8 kódolás.
- A megoldást tartalmazó osztálynak a `game.mario.MarioPlayer`-ből kell származnia, ami a keretrendszer részét képezi
- Véletlen számok használata esetén kizárólag az örökölt `random` mezőt szabad használni, és a seed átállítása tilos
- A megoldást tartalmazó osztálynak részletes magyar osztálydokumentációt kell tartalmaznia, javadoc formátumban, illetve a kód dokumentációja is magyar kell, hogy legyen
- A kód nem használhat a keretrendszeren kívül semmilyen más osztálykönyvtárat (természetesen a JDK osztályain kívül)
- A megoldást tartalmazó osztály nem lehet csomagban
- A megoldásban nem lehet képernyőre írás
- A megoldás nem nyithat meg fájlt, nem indíthat új szálát
- Az implementált metódusoknak minden esetben vissza kell térniük (nem szerepelhet benne `exit` hívás például)
- A forráskód első sorában megadható egy nicknév és egy értesítési emailcím a következő formátumban:

```
///Nicknevem,Vezeteknev.Keresztnev@stud.u-szeged.hu
```

Ha meg van adva, a nicknév jelenik meg a ranglistában, egyébként pedig a Neptun azonosító. Ha meg van adva emailcím, egy tájékoztató emailt küldünk az ágens kiértékelése után, mely a `{DATE}_out.txt` (a program kimenete), `{DATE}_log.txt` (játék logja), és `meta.txt` (eddig beküldések státusza) állományok elérhetőségét tartalmazza. Emailcím megadása nélkül is megtekinthető a ranglistában a pontszám és a játék visszajátszható. Lehetőség van arra is, hogy nicknevet ne, csak emailt adjunk meg, ebben az esetben az első paramétert üresen kell hagyni, majd a vessző után az emailcímet megadni:

```
///,Vezeteknev.Keresztnev@stud.u-szeged.hu
```

Az email értesítő esetén érdemes hivatalos egyetemi emailcímet használni. (A gmail pl. spam folderbe teheti az értesítést.)

- Fenntartjuk a jogot, hogy bármilyen, fent nem listázott, de az etika szabályai ellen történő vétséget szankcionáljunk; ha bárkinek kételyei vannak egy konkrét dologgal kapcsolatban, inkább kérdezzen rá időben.