

Tárgy: BPI1235L *Mesterséges intelligencia*

Oktató: Vályi Sándor

Kreditpont : 4

Kontakt óraszám (elm.+gyak.): 9+9

Félévi követelmény: vizsga

Előfeltétel (tantárgyi kód) BPI1204 (programozás 2)

Évközi követelmények, vizsgára bocsájtás feltételei:

1. A PTI-hallgatók egy MI feladatot számítógépes **program készítésével** megoldanak és dokumentálva benyújtanak. Ez kritérium-feltétel, pontszám nem tartozik hozzá. Határidő: az utolsó évközi gyakorlat.

2. papíron megírt gyakorlati feladatokból álló **zárhelyi dolgozat**, az utolsó évközi gyakorlaton. Ez 80 pont.

Amennyiben valamelyik (akár mind2) évközi követelmény nem sikerül, úgy ezeket 1x lehet javítani a vizsgaidőszak közepén, a TVSZ-nek megfelelően.

Vizsga: Papíron megírt vizsga elméleti kérdésekkel. Ez a vizsga 20 pont. Ebből a vizsgából 3 lesz, a TVSZ-ek megfelelően, az egyik az évközi gyakorlati követelmények pótlása után közvetlenül, ugyanazon a napon. Ha egy vizsgát felvesz a hallgató, úgy, hogy nincs meg az évközi követelmény, úgy elégtelent fogok beírni. Ha az évközi követelmény teljesült, de nem jön el a hallgató, akkor azt a dolgozatot 0 ponttal beszámítom, és lehet vizsgajegyet szerezni.

Az elégségeshez 50 pont kell, az évközi teljesítményből (program, 80p) és a papíron írt vizsgából (20p) összegezve. Minden 10% javulás egy jegy javulást jelent, maximum ötösig.

A beadandó program az **állapottér gráfrepresentációján** kell alapuljon és valamely, a kurzuson **tanult keresőalgoritmust** felhasználva (mélységi, szélességi kereső etc. avagy minimax algoritmus, alfa-béta vágás) kell működnie.

1-személyes játék esetén kezdődjön pályaszerkesztővel, ha van értelme, 2-személyes játék esetén is kezdődjön pályaszerkesztővel (állászerkesztővel). 1 személyes játék esetén a megszerkesztett feladatra fusson a kereső eljárás, és lépésenként mutassa be a megtalált megoldást. 2 személyes játék esetén az egyik ellenfél lépéseit humán vezesse, a másik fél lépéseit a minimax lépésajánló, legalább 8 féllépés mélységig.

Mindkét esetben a beadandó program **grafikus felhasználói felülettel** rendelkezzen. Legalább **annyi távoli adatbázis-használatot** tartalmazzon, hogy elmenti az adott felhasználó nevét, a megszerkesztett generált pályát, a játszmát és az eredményt, vagy relációs adatbázisban, vagy XML-adatként.

A megoldás **elkészítése előtt** a tanárral való **egyeztetés szükséges** a megírandó programmal szemben támasztott **követelményekről**. Ennek módja a gyakorlatokon való személyes megjelenés. A program megoldó algoritmus az állapottér-reprezentáción kell alapuljon, s valamelyik, az előadáson vett kereső-algoritmust kell megvalósítania.

- A bemutatók feladatokban való mély ismereteikről **személyes védés** során számolnak be. Az elkészített programban meg kell jelölni azon kódrészeket, amelyeket nem a védő készített. Olyan kérdések is várhatók, hogy mit kellene másképp csinálni, ha változtatni akarnánk a program működésén. Amennyiben a védés nem sikeres, a vizsgázást nem engedélyezem.
- A **dokumentáció** tartalmazza az állapottér egy 10-12 állapotú részének ábráját, ahol a lehetséges operátorok is elmagyarázhatók, valamint azt, hogy milyen kereső lett leprogramozva. A program az állapottérben való keresés egyes lépéseit (az érintett csúcshalmazokat és az alkalmazott műveletet) is tudja kiírni.

Kivétel a fent leírt beadandó-követelmények alól:

- aki tanuló ágenst készít open source könyvtárral adott inputból adott intervallum-értékeket előállító intervallum-értékű számítások előállítására, pl. 'Deep Learning' neuronhálókat alkalmazva. Neki erre a beadandóra kapható pontjainak száma 90p.
- aki működő ágenst készít reális Sokoban feladatok megoldására. Neki erre a beadandóra kapható pontjainak száma 90p.
- aki olyan webhelyet készít, Apache+PHP, Node.js vagy Apache Tomcat, Spring MVC (ez a preferált!) technológiára, amelyre a regisztrált userek felvihetnek Sokoban álláskiértékelő heurisztikát visszaadó int heur(SokobanState) metódus szövegét, Java-ban megírva, és a webhely eme heurisztikát injektálva az A* keresőbe keres megoldást a szintén a userek által felvitt pályákra, és időszakonként rangsort hirdet a userek heurisztikái között, a teljesítményük alapján.

Tananyag:

Hét	Előadáson	Gyakorlaton	
1	Bevezetés, intelligens ágensek	Ágensek tulajdonságai; Egyszerű ágensek programozása.	
1	Problémák állapottér-reprezentációja	Példák állapottér-reprezentációra, keretrendszer állapottér-reprezentált problémák számára	
1	Problémamegoldás, mint útkeresés	Mélységi, szélességi és egyéb nem informált keresők.	
2	Informált keresési eljárások	Informált keresők.	
3	2-személyes játékok	Minimax algoritmus, alfa-béta vágás. Keretrendszer a 2-személyes játékokhoz. Példa a beadandóhoz.	

4	<p>Genetikus algoritmusok, kényszerfeltételekkel definiált problémák megoldása. Logikai ismeretreprezentáció, tételbizonyító algoritmus (rezolúció), logikai programozás. Szakértői rendszerek, bizonytalan információ ábrázolása. Tanuló algoritmusok.</p> <p>Papír alapú dolgozat I a gyakorlati részekből.</p> <p>(anyag: Ágensek, állapottér-reprezentáció, útkereső algoritmusok, 2-személyes játékok, minimax, alfabéta-vágás).</p>	<p>Logikai ismeretreprezentáció</p> <p>Tarski világában és a wumpuszvilágban. Prolog programok írása, használata Java-ból. Bayes-hálók. ID3-algoritmus.</p>	
5	<p>Papír alapú dolgozat II a gyakorlati részekből.</p> <p>(Anyag: kényszerfeltételekkel definiált problémák felírása, logikai ismeretreprezentáció, logikai programozás, rezolúció, Bayes-háló, ID3-algoritmus).</p>	<p>A beadandók védeése.</p>	

Oktatási segédanyag:

Az előadáson bemutatott prezentációk megtalálhatók a <http://moodle.nye.hu> címen.

Kötelező¹ és ajánlott irodalom:

- Futó Iván (szerk.): Mesterséges intelligencia, Aula Kiadó, 1999.
- Fekete István, Gregorics Tibor, Nagy Sára: Bevezetés a mesterséges intelligenciába, LSI Oktatóközpont, 1990, ELTE Eötvös Kiadó, 2006.

Stuart J. Russell, Peter Norvig¹: Mesterséges intelligencia modern megközelítésben, Panem, Budapest, 2005. (1-9. fejezet a kötelező, a 2003-as második kiadás fordítása) ,
Online elérhető, URL:

<https://dtk.tankonyvtar.hu/handle/123456789/8703>