

## **BPI1235 Mesterséges intelligencia**

### **Tantárgyi tematika és félévi követelményrendszer**

**Kreditpont : 4**

**Heti kontakt óraszám (elm.+gyak.): 2+2**

**Előfeltétel (tantárgyi kód) BPI1204 (programozás 2)**

***Oktató: Vályi Sándor***

- *elektronikus levelezőcím: [valyi.sandor@nye.hu](mailto:valyi.sandor@nye.hu), 2 munkanapi reakcióidő, nem azonnali ügyfélszolgálat!*
- *A kurzushoz tartozik: [moodle.nye.hu](https://moodle.nye.hu) (Kezdőoldal / ► Matematika és Informatika Intézet / ► Informatika / ► PTI III. év nappali / ► Mesterséges intelligencia). Itt lesznek a jelenléti előadások anyagai.*
- *A kurzushoz tartozik: [classroom.google.com](https://classroom.google.com)- Felvételi kulcs: f6nc2nj itt fogunk gyakorló feladatokat megoldani interaktívan, de jelenléti módon, a tanteremben.*

**Félévi követelmény:** vizsga (évközi alapján lehetséges jegymegajánlás is)

**Évközi követelmény, vizsgára bocsájtás feltétele:**  
beadandó-védés és 2 dolgozat

**Féléves tematika:**

### **Az órák programja (tematika):**

<b>Hét</b>	<b>Előadáson</b>	<b>Gyakorlaton</b>	<b>Dátum</b>
<b>1</b>	Bevezetés, intelligens ágensek	Ágensek tulajdonságai; Egyszerű ágensek programozása	<b>febr. 23</b>
<b>2</b>	Problémák állapottér-reprezentációja	Példák állapottér-reprezentációra, keretrendszer állapottér-reprezentált problémák számára	<b>márc.2</b>
<b>3</b>	Problémamegoldás, mint útkeresés	Mélylési, szélességi és egyéb nem informált keresők.	<b>márc.9</b>
<b>4</b>	Informált keresési eljárások	Informált keresők	<b>márc.16</b>
<b>5</b>	2-személyes játékok, Java keretrendszer a 2-személyes játékokhoz. Példa a beadandó 0-1-2. produktumára	Minimax algoritmus, alfa-béta vágás. Beadandó-témaválasztás.Beadandó 0. feladat.	<b>márc.23</b>

6	Adatbázist nem építő keresési eljárások. Genetikus algoritmusok, backtrack eljárás. Kényszerfeltételek az állapotokon.	Beadandó 1. produktum, példa dolgozat	márc.30
7	ZH-készülés	<b><u>ZH I.</u></b>	ápr.6
8	Beadandó példaprogram bemutatása	Beadandó 2. produktum	ápr.13
9	Nulladrendű és elsőrendű logika és ismeretreprezentáció, következtetés, automatikus következtetés: rezolúció ítéletlogikára	Ismeretreprezentáció Tarski világában. Logikai feladatok megoldása -- tutorial.	ápr.20
10	Logikai programozás, Prolog, kényszerfeltételes logikai programozás. Szakértői és döntéstámogatási rendszerek. Ismeretreprezentációs módszerek. Szakértői rendszerek és logikai programozás.	Prolog gyakorlatok. Beadandó 3. fázis	ápr.27
11	Bizonytalan ismeretek ábrázolása, pl. Bayes-hálók. Tanuló algoritmusok, Döntési fák.	Gyakorlatok a Bayes-hálókról, ID3 algoritmusról, döntési fákról	máj.4
12	Neuronhálók, deep learning a szövegfeldolgozásban	Beadandó 4. fázis:	máj.11
13	<b>ZH II. dolgozat az előadáson</b>	Beadandó-bemutató és védés hosszabbításra.	máj.19
14	Tartalék időpont	Tartalék időpont	máj.26

## ***Értékelési rendszer (évközi és vizsga)***

-- **évközi 1. követelmény:** 40 pont **beadandó**, ami a későbbiekben nagyon részletesen le van írva. Ennek benyújtása messze nem elég: védése a gyakorlatokon.

A beadandóválasztás az ötödik hétig a moodle-n. A beadandó köztes állapotainak bemutatása szintén, a gyakorlatokon. Benyújtási határidő a táblázatban. Minimum pontszám 24, anélkül nem lehet vizsgázni.

Fázis	Leírás	Hét	Pontszám
0. fázis	a feladat szöveges leírásának feltöltése	5	2

1. <i>fázis</i>	a választott témák problémáinak állapottér-reprezentációja, állapotok, operátorok leírása dokumentációban	6	4
2. <i>fázis</i>	állapottér-repr. Javában programozva is, kereső-algoritmus beállítása	8	15
3. <i>fázis</i>	GUI pályaszerkesztés	10	15
4. <i>fázis</i>	mentés XML-be/RDBMS-be, visszatöltés is, természetesen	12	4

-- **évközi 2. item:** 40 pont *elméleti és gyak dolgozat papíron, a 7. heti előadáson + a 7. heti gyakorlaton.* Minimum 24 pont, anélkül nem lehet vizsgázni.

-- **évközi 3. item,** 20 pont: *gyak dolgozat II papíron, a 13. heti előadáson.* Minimum 10 pont a vizsgázáshoz.

Akinek eme évközi 3 követelményből megvan 50p a max 100-ból, **ő vizsgázhat.**

**Jegymegajánlás:** az évközi teljesítmény alapján 55p – elégséges, 70 p – közepes, 80p- jó, 90 p – jeles.

Az **évközi követelményeket egyszer lehet csak ismételni várhatóan június elején**, ezek alapján már nem kötelező a jegymegajánlás az oktató számára. De lehetséges.

**A vizsga: szóbeli.** Itt a jegymegajánlás és az évközi pontok már nem számítanak, minden jegyet lehet szerezni 1..5 intervallumban.

#### **A beadandóról:**

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

A problémamegoldás/játék inicializálható legyen egy kezdőállással, amit input fájlból olvasunk föl. 2 személyes játék esetén az egyik ellenfél lépéseit humán vezesse, a másik fél lépéseit a minimax lépésajánló, legalább 8 féllépés mélységig.

Előny, ha a program grafikus felhasználói felülettel rendelkezik, pályaszerkesztési funkcióval (a kezdőállás grafikus megszerkesztése). Legalább annyi távoli adatbázis-használatot tartalmazzon, hogy elmenti az adott felhasználó nevét, a megszerkesztett generált pályát egy megadott névvel, a játszmát és az eredményt, vagy relációs adatbázisban, vagy XML-adatként. Mondani sem kellene, vissza is kell tudnia tölteni a mentésből a név alapján kikeresve a több elmentett állás közül.

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 állapotter (játékfa) egy 10-12 állapotú részének ábráját, ahol a lehetséges operátorok is elmagyarázhatók, valamint azt, hogy milyen kereső lett leprogramozva. A program az állapotterében való keresés egyes lépéseit (az érintett csúcshalmazokat és az alkalmazott műveletet) is tudja kiírni. [12p]

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, ő jeles megajánlott jegyet kap, s sejtetően mehet vele országos TDK-ra is.
- aki működő ágenst készít reális (amely problémákat szokták mellékelni a PC-s sokobanhoz) Sokoban feladatok megoldására, jeles megajánlott jegyet kap.
- aki olyan webhelyet készít, pl. Java web service, 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, ő jeles megajánlást kap.

### Az érdemjegy kialakításának módja:

*A teljesítménnyel pozitív korrelációban, a pontszámok alapján..*

### Oktatási segédanyag:

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

### Kötelező<sup>1</sup> 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 Norvig<sup>1</sup>: 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>

