



# KOOPERÁCIÓ ÉS GÉPI TANULÁS LABORATÓRIUM

## Szavazás és Aukció Gyakorlati feladatok

Készítette:  
*Kovács Dániel László*  
([dkovacs@mit.bme.hu](mailto:dkovacs@mit.bme.hu))

Méréstechnika és Információs Rendszerek Tanszék  
Budapesti Műszaki és Gazdaságtudományi Egyetem

2012, október.

### KÖVETELMÉNYEK:

- A laborgyakorlat teljesítésére 4 órányi tiszta munkaidő áll rendelkezésre.
- A laborgyakorlat megkezdéséhez maradéktalanul el kell készíteni az otthoni felkészülési feladatokat, és azok dokumentációját a jegyzőkönyvben (amit a laborra is hozni kell).
- A laborgyakorlat időre történő, maradéktalan teljesítéshez a segédanyagok hiánytalan ismeretét feltételezzük.
- A segédanyagok ismeretét egy 5 perces „beugró” keretében mérjük fel a gyakorlat elején.
- A sikeres teljesítéshez a gyakorlat során végzett munka – a futási eredményeket is beleértve – jegyzőkönyv formájában történő dokumentációja és határidőre történő leadása szükséges.

### JEGYZŐKÖNYV:

A jegyzőkönyvnek (amiben már szerepel az otthoni felkészülési feladatok dokumentációja, és amit nyilván a laborgyakorlatra is magával kell hoznia akár pendrive-on, akár CD-én, vagy akár a webről letöltve (pl. önmagának elküldött email-hez csatolva, majd onnan letöltve a laborgépre)) – a megoldó(k) nevén és neptun-kódján kívül – a következőket kell tartalmaznia minden egyes részfeladatra vonatkozóan:

- KI-MIT oldott meg (feladat felosztása és értelmezése, pl. bővített informális leírás, stb).
- HOGYAN oldották meg (megoldás leírása, pl. PDDL változtatások magyarázata, stb).
- MIÉRT így oldották meg (megoldás indoklása).
- EREDMÉNYEK (bemutatása, értelmezése és értékelése)

### LEADÁS:

A teljes \jade\src\msclab01\votingauction\_lab könyvtárat ZIP-pelve, illetve a **jegyzőkönyvet** MS Word 2003-as DOC, DOCX, vagy PDF formátumban **egyetlen ZIP-be tömörítve** kell a laborgyakorlaton megjelölt **határidőig** beadni a <https://hf.mit.bme.hu> weboldalon.

**A nem megfelelő formában beadott anyagok értékelés nélkül elutasításra kerülnek!!!**

## OTTHONI FELKÉSZÜLÉSI FELADATOK (KÖTELEZŐ)

1. A laborsegédlet és az egyéb segédanyagok maradéktalan elolvasását és megértését követően töltsse le a labor weblapjáról (<http://www.mit.bme.hu/oktatas/targyak/vimim223/feladat/5-Szavazas-es-aukcio>) a labor forráskódjait, és bontsa ki a forráskódokat a \jade\src\msclab01 könyvtárba. Ezt követően indítson el Eclipse alól egy JADE platform-ot, majd legalább három msclab01.votingauction\_lab.VoterAgent.VoterAgent szavazó ágenszt (pl. va0, va1, és va2 néven) a \jade\src\votingauction\_lab\cfg\voting01.cfg szavazási konfigurációnak megfelelő – akár egyező, akár különböző – jelöltekkel, illetve egy msclab01.votingauction\_lab.VotingMechanismAgent.VotingMechanismAgent szavazásvezérlő ágenszt (pl. vma néven) az előbb említett konfigurációval.
  - a. Figyelje meg és értelmezze a működést.
  - b. Kövesse nyomon Sniffer ágenssel az ágensek közti üzenetcserét.
  - c. Mit tapasztal?
  - d. Más-más szavazó ágens paraméterek megadása hatására hogy viselkedik a rendszer?
  - e. Milyen szabályok szerint zajlik a szavazás?
  - f. Mi az előnye, és mi a hátrulatója ennek a fajtájú szavazásnak?
  - g. A lefutás során/után milyen adatok jöttek létre a \jade\src\votingauction\_lab\log könyvtárban?
2. Tegye több (legalább két) körös runoff szavazássá az előbbi protokollt! Ehhez megfelelőképp írja át a VotingMechanismAgent szavazásvezérlő ágens `maxRounds` változóját, majd tesztelje az előbbi feladathoz hasonlóan a rendszert. *Megjegyzés: többkörös szimpla többségi runoff szavazás esetén, ha a szavazás egy adott körében nem születik egyértelmű végeredmény, azaz nincs egy opció, amely a beérkezett szavazatok alapján maximális minősítésű, úgy a több maximális minősítésű opcióval újabb szavazás kerül kiírásra.*
3. Alakítsa át a szavazást **többségi elven működő szavazássá (majority rule voting)**! Ehhez egészítse ki a VotingMechanismAgent ágens `AnnounceAndWait` viselkedésének `action()` metódusában, a `case 1` részben (mikor `state=1`, azaz az ágens szavazatokra vár) a megfelelő két feltételvizsgálatot (`if(CommonMethods.maxNum(votes) == 1)`) úgy, hogy ne csak azt nézzék, hogy egyetlen maximális minősítésű opció van-e az eddig beérkezett szavazatok alapján, hanem azt is, hogy a szavazatok száma erre az opcióra legalább az ágensek számának fele-e, vagy sem. Tesztelje a módosított szavazási protokollt az előbbi feladatokhoz hasonlóan, és foglalja össze, ill. magyarázza meg a különbségeket és hasonlóságokat!
4. Térjünk most át az aukciókra! Indítson el legalább három msclab01.votingauction\_lab.BidderAgent.BidderAgent licitáló ágenszt (pl. ba0, ba1, és ba2 néven) a \jade\src\votingauction\_lab\cfg\auction01.cfg aukció konfigurációval, illetve egy msclab01.votingauction\_lab.AuctioneerAgent.AuctioneerAgent árverező ágenszt is (pl. aa néven) ugyanezzel.
  - a. Figyelje meg és értelmezze a működést.
  - b. Kövesse nyomon Sniffer ágenssel az ágensek közti üzenetcserét.
  - c. Mit tapasztal?
  - d. Milyen szabályok szerint zajlik az aukció?
  - e. Mi az előnye, és mi a hátrulatója ennek a fajtájú aukciónak?
  - f. A lefutás során/után milyen adatok jöttek létre a \jade\src\votingauction\_lab\log könyvtárban?
  - g. Milyen licitálási stratégia szerint játszik most a BidderAgent ágens?
  - h. **Szorgalmi feladat:** próbáljon meg javítani a BidderAgent ágens licitálási stratégiáján, és magyarázza meg, hogy mit és miért csinált!

## LABORBAN MEGOLDANDÓ FELADATOK

A következő feladatok megoldását az otthoni felkészülési munka során előállt anyagokra alapozva kell folytatnunk **(EZEKET AZ ANYAGOKAT A JEGYZŐKÖNYV-KEZDEMÉNNYEL EGYÜTT MINDENKÉPP HOZZUK MAGUNKKAL A LABORRA!!!)**

1. Alakítsa át a `VoterAgent` szavazó ágenszt úgy, hogy ne csak egy, hanem több opcióra is képes legyen egy-egy körben szavazni (ehhez akár a `VotingMechanismAgent` ágens szavazási konfigurációjához hasonló konfigurációt is létrehozhat számára, vagy átadhatja a több óhajtott opciót bemeneti paraméterként, ha egyszerűbb, vagy lehet akár véletlenszerű is, hogy mire szavaz az ágens)! Ehhez a `ParticipateInVoting` viselkedés `case 1` esetét kell módosítani. Ennek kapcsán ne felejtse el kivenni a `VotingMechanismAgent` ágens `AnnounceAndWait` viselkedésének `action()` metódusában a `case 1` rész `for` ciklusából a `break`-et. Tesztelje a működést! Mit tapasztal? Hogyan zajlik ez az újfajta, úgynevezett **engedélyező szavazás (approval voting)**? Mi a különbség, hasonlóság, előny, hátrány az előzőleg megvalósított/kipróbált módszerekhez képest?
2. Az előbbiekhöz hasonlóan módosítsa most a `VoterAgent` szavazó ágenszt úgy, hogy adott határok közt pontozni is tudja az egyes opciókat (ne csak binárisan szavazzon)! Tesztelje az így létrejött újfajta, úgynevezett **pontozó szavazást (rated voting)**! Mi a különbség, hasonlóság, előny, hátrány az előző módszerekhez képest?
3. Az előbbiekhöz hasonlóan módosítsa most még tovább a `VoterAgent` szavazó ágenszt úgy, hogy már rangsorolni is tudja az opciókat ( $0$ -tól  $(k-1)$ -ig, ha  $k$  darab opció van)! Tesztelje az így létrejött újfajta, úgynevezett **Borda-féle rangsor alapú számlálást (Borda count)**! Mi a különbség, hasonlóság, előny, hátrány az előzőekhez képest?
4. Most pedig térjünk át az aukciókra! Hozzon létre **japán aukciót**! Ehhez például átírhatja a `BidderAgent` licitáló ágensek licitálási stratégiáját úgy, hogy egy adott összeg fölött „szálljanak ki az aukcióból”, azaz ne növeljék tovább a mások által megemelt tétet, de még szebb megoldás, ha külön üzenetváltást vezetünk be erre a célra a `BidderAgent` licitáló ágensek, és az `AuctioneerAgent` árverező ágens között. Pl. a licitáló küldhet egy `DISAGREE` performatívájú üzenetet az árverezőnek („kimehet a szobából”), mire az árverező kiveheti őt az aukcióból, és ezt a többi/maradék licitálónak is a tudomására hozhatja pl. egy megfelelő `INFORM` üzenetben. Teszteljük, és értelmezzük a működést! Helyesen zajlik az aukció? Jobb-e, vagy rosszabb, mint az előző? Miért?
5. Alakítsa át az előbbi aukciót **holland aukcióvá**! Ehhez írja át az `AuctioneerAgent` árverező ágens `AnnounceAndWait` viselkedésének `action()` metódusa `case 1` esetét (úgy, hogy licit esetén azonnal menjünk át `state=2`-be!), illetve írjuk át megfelelőképp a „kalapács leütéséért” felelős `TimeoutHandler` osztály `onWake()` metódusát is (tájékozódjunk a bő kommentek alapján). Érdemes felfigyelnünk az aukció végét jelző harmadik leütésre, mikor az `if(round == 3)` feltételvizsgálat teljesül. Ez a feltételvizsgálat a holland aukció esetén nyilván inkább valami olyasmi kellene, hogy legyen, ami azt vizsgálja, hogy egy-egy termék kapcsán egy adott minimális ár alá mentünk-e már, vagy sem. Ha igen, akkor vége az aukciónak (hiszen a termék nem kelt el még a minimális áron sem), egyébként nincs vége. Azt is gondoljuk meg, hogy a `BidderAgent` ágensek részéről érkező licitek ekkor már nem az árverező által megajánlott összeg fölött kellene, hogy legyenek, hanem azzal egyenértékűek. Viszont ezt nyilván az `AuctioneerAgent` árverező ágensnél is át kell vezetni (`proposal == bid`)... Ha elkészültünk, teszteljük, és értelmezzük a működést! Helyesen zajlik az aukció? Jobb/rosszabb, mint az előzőek? Miért?
6. **Szorgalmi feladat:** alakítsa át valamely előbbi aukciót **szekvenciális aukcióvá**!