



## Други пројектни задатак из предмета Интелигентни системи школске 2021/22. године

### Основне информације

**PyStolovina** је графичка симулација секвенцијалне игре написана на програмском језику Пајтон. Главни прозор апликације приказује дводимезионалну мапу поља по којој се крећу један студентски и бар један наставнички агент. Поље може да буде пут или рупа. Агенти се крећу наизменично по једно поље хоризонтално, вертикално или дијагонално у једном потезу, и то искључиво по путевима. Након преласка на наредно поље пута, претходно поље пута постаје рупа. Циљ сваког агента јесте да онемогући кретање свим осталим агентима.

### Начин коришћења апликације

Програм се покреће из терминала навођењем следеће команде:

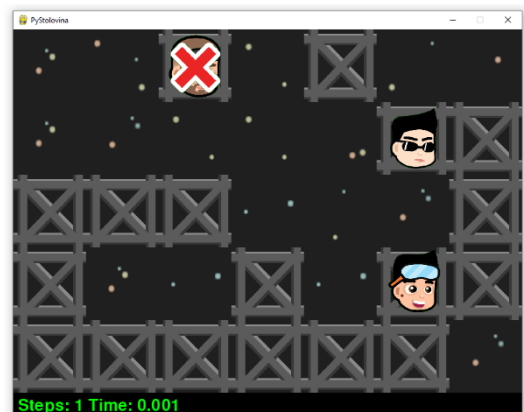
```
.\main.py map agent time_to_think max_levels
```

где су:

- `.\main.py` – путања до главног Пајтон фајла са изворним кодом
- `map` – релативна путања до текстуалног фајла са конфигурацијом мапе (подразумевано `maps\map0.txt`)
- `agent` – назив класе студентског агента који се користи (подразумевано `StudentAgent`)
- `time_to_think` – целобројни параметар који представља максимално време у секундама за доношење одлуке (акције агента) (подразумевано 1 секунда)
- `max_levels` – целобројни параметар који представља максималну дубину развијања стабла игре (подразумевано -1, што означава претрагу без ограничења)

Пре покретања потребно је инсталирати пакет `pygame` (верз. 2.1.0) у оквиру Пајтон интерпретера. Након покретања приказује се главни прозор апликације.

У оквиру прозора приказана је мапа поља по којој се агенти крећу, испод које се налази секција са информацијама о тренутном потезу игре и протеклом времену размишљања агента који је на потезу. Притиском на дугме `SPACE` могуће је покренути и привремено зауставити кретање агента. Притиском на дугме `ESC` могуће је прекинути рад апликације и затворити њен главни прозор.



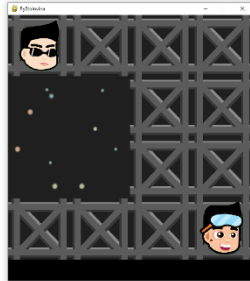


## Mana

Мапа је текстуална датотека која у произвољном броју редова исте дужине садржи ознаке поља и агената (видети ниже) која чине мапу.

Пример фајла мапе:

```
Orrr  
hrrr  
hrrr  
rrrl
```







На мапи се могу наћи поља дата у табели:

Назив	Слика	Ознака у мапи	Проходно
Пут		r	да
Рупа		h	не

## Постојећи агенти система

У систему постоје имплементирани следећи наставнички агенти:

- Aki. Агент у сваком потезу бира следећу акцију која га води до поља које је најближе студентском агенту по Менхетн дистанци, а у случају две или више таквих акција бира акцију по страни света (север, североисток, исток, југоисток, југ, југозапад, запад, северозапад). Агент се у мапи наводи ознаком 1. 
- Jocke. Агент у сваком потезу бира следећу акцију насумично. Агент се у мапи наводи ознаком 2. 
- Draza. Агент користи студентску имплементацију Минимакс алгоритма са алфа-бета одсецањем и може да се користи у сценаријима тек када је имплементација завршена. Агент се у мапи наводи ознаком 3. 
- Bole. Агент користи студентску имплементацију генерализације Минимакс алгоритма за игре са више од два играча и може да се користи у сценаријима тек када је имплементација завршена. Агент се у мапи наводи ознаком 4. 



## Кориснички захтеви

Студенти треба да имплементирају алгоритме секвенцијалних игара у изведеним класама класе `StudentAgent` редефинисањем њене функције `get_next_action`, чија је повратна вредност једна од акција из модула `actions.py` (`NORTH`, `NE`, `EAST`, `SE`, `SOUTH`, `SW`, `WEST`, `NW`) и представља следећу акцију коју ће агент извршити. Студентима је у оквиру кода класе `StudentAgent` дат пример имплементације наведене функције која на случајан начин бира наредну акцију. Студентски агент се у мапи увек наводи ознаком `0`. Класе чије је методе потребно имплементирати налазе се у `students.py` модулу и то су:

- *MinimaxAgent*. Агент представља реализацију Минимакс алгоритма за два рационална играча.
- *MinimaxABAgent*. Агент представља реализацију Минимакс алгоритма за два рационална играча уз коришћење алфа-бета одсецања.
- *MaxNAgent*. Агент представља реализацију генерализације Минимакс алгоритма за игре са више од два рационална играча у којој сваки играч игра само за себе (не постоје савези између играча).
- *ExpectAgent*. Агент представља реализацију Експектимакс алгоритма за два играча који се састоји искључиво из *Max* и *Chance* (сматра се да сви противници играју насумично) чворова.

У случају да је за студентског агента пораз неизбежан (по свим листовима стабла), следећа акција се бира по странама света (север, североисток, исток, југоисток, југ, југозапад, запад, северозапад). На располагању је пет мапа које студенти могу да користе за проверу свог решења:

- *map0.txt*, *map1.txt*, *map2.txt* - Основне мапе за тестирање Минимакс агената са и без алфа-бета одсецања.
- *map3.txt* - Основна мапа за тестирање генерализације Минимакс алгоритма за игре са више од два играча.
- *map4.txt* - Основна мапа за тестирање Експектимакс агената.

Студентима се предлаже да провере решење са различитим временима за доношење одлуке и максималним дубинама за развијање стабла и упореде резултате и перформансе. Студентима се такође предлаже да имплементирају неколико различитих функција процене и размисле о предностима и манама сваке. Имати у виду да се на одбрани домаћег задатка могу појавити нове мапе и нови агенти. Имплементација се ради у програмском језику Пајтон уз дозвољено коришћење свих стандардних библиотека и структура података.



Универзитет у Београду - Електротехнички факултет  
Катедра за рачунарску технику и информатику

### Напомене

Електронску верзију решења овог домаћег задатка послати најкасније до 20. новембра 2021. године до 15 часова, као ZIP архиву на следећем [линку](#). Одбрана домаћег задатка планирана је 21.12.2021. године.

Пројектни задатак из предмета *Интелигентни системи* се ради самостално и није обавезан за полагање испита (на испиту се може заменити са теоријским питањима из целокупног градива). Овај пројектни задатак се може бранити само у за то предвиђеном термину. Пројектни задатак вреди максимално 10 поена (уз могућност додатних поена за радове који се посебно истакну).

На усменој обрани кандидат мора самостално да покрене своје решење. Кандидат мора да поседује потребан ниво знања о задатку, мора да буде свестан недостатака приложеног решења и могућности да те недостатке реши. Кандидат мора тачно да одговори и на одређен број питања која се баве тематиком домаћег задатка.

Евентуална питања послати асистентима на мејл, али као једну поруку, а не две одвојене (другог асистента обавезно ставити у копију - СС поруке).

aki@etf.rs

jocke@etf.rs