



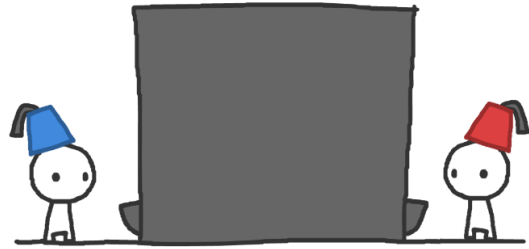
## Први пројектни задатак из предмета Интелигентни системи школске 2020/21. године

### Основне информације

**Split or steal** је симултана игра за два играча која се заснива на концептима дилеме затвореника са измењеним правилима. Играчима се нуди заједничка награда коју могу да поделе на два једнака дела. Додатно, сваки играч има право да се уместо поделе одлучи за крађу целокупне награде. У случају да се један играч одлучи за крађу награде, а други играч за дељење награде, целокупну награду односи играч који је одабрао крађу. У случају да оба играча одаберу крађу, ниједан играч не добија награду.

### Правила игре

**Cooperate or cheat** је игра заснована на игри **Split or steal**. Играче дели машина за мултипликацију новчића тако да они не виде један другог. Да би машина извршила мултипликацију новчића, потребно је убацити један или два новчића. Сваки од играча може да одабере да убаца један, два новчића или да не ради ништа. Машина на супротном крају од краја где је убачен новчић



производи одређен број копија новчића. Ако је убачен један новчић, машина производи три новчића. Ако су убачена два новчића, машина производи седам новчића. Играчи су у дилеми да ли сарађивати и додавати новчиће што производи добитак за супротног играча у нади да ће и други играч радити исто или преварити другог играча и не губити непотребне сопствене новчиће на његов добитак. Матрица добитака ове игре приказана је у наставку.

	Не убацуј новчић	Убаца 1 новчић	Убаца 2 новчића
Не убацуј новчић	(0, 0)	(3, -1)	(7, -2)
Убаца 1 новчић	(-1, 3)	(2, 2)	(6, 1)
Убаца 2 новчића	(-2, 7)	(1, 6)	(5, 5)



## Опис система

Опис коришћених класа система дат је у [документацији](#). Систем се дистрибуира студентима као *.jar* фајл који може да се покрене. Покретањем *.jar* фајла започиње се симулација игре чија је конфигурација задата кроз први параметар командне линије којим се наводи путања до конфигурационог фајла. Конфигурациони фајл је текстуални фајл чији је формат дат у документацији. Студентима су на располагању конфигурациони фајлови свих јавних сценарија. Конфигурационе фајлове је потребно изменити додавањем линије са називом класе имплементираних студентског агента и бројем агената у игри, који мора да одговара броју осталих наведених агената. Подаци у једној линији датотеке одвојени су зарезом. Други параметар командне линије представља путању до излазног фајла у којем ће резултат симулације бити исписан. Игра се састоји из симулације мечева између свака два играча у игри. Сваки меч се састоји из одређеног броја рунди (потеза) задатих конфигурационим фајлом. Играчу су током меча у сваком тренутку доступни дотадашњи потези противника. Након сваког одиграног меча врши се ресетовање играчевог стања и у излазни фајл се исписује резултат меча. Након последњег меча у излазни фајл се исписује резултат игре.

## Агенти система

У систему постоје следећи агенти, сваки са својом стратегијом:

- *Goody*. Агент увек игра стратегију „убаци два новчића“.
- *Stinger*. Агент увек игра стратегију „не убацуј новчић“.
- *Copycat*. Агент у првом потезу игра стратегију „убаци један новчић“. Сваки следећи потез игра копирајући претходни потез противника.
- *Forgiver*. Агент у првом потезу игра стратегију „убаци један новчић“. Сваки следећи потез зависи од претходних потеза противника. Уколико је противник у претходном потезу играо стратегију „убаци 1 новчић“ или „убаци 2 новчића“, агент игра копирајући такав потез противника. Уколико је противник у претходном потезу одиграо стратегију „не убацуј новчић“, агент наставља да игра као што је играо у претходном потезу. Уколико је противник у последња два потеза одиграо „не убацуј новчић“, агент копира такав потез противника.
- *Avenger*. Агент у првом потезу игра стратегију „убаци два новчића“. Уколико је противник у претходном потезу играо стратегију давања мањег броја новчића од агента, онда агент копира стратегију противника из претходног потеза.



## Кориснички захтеви

Потребно је имплементирати изведену класу класе *Player* тако да њени агенти побеђују у сваком од следећих сценарија:

- Сценарио А. Игру чини по један агент сваке од доступних изведених класа класе *Player*. Игра се завршава након што свака два играча одиграју по један меч.
- Сценарио В. Игру чине по N агената сваке од доступних изведених класа класе *Player*. Игра се завршава након што свака два играча одиграју по један меч.
- Сценарио С. Игру чине по N агената сваке од доступних изведених класа класе *Player*. Након што свака два играча одиграју по један меч, X најгоре пласираних играча се избацује из игре и игра се понавља док год постоје барем два играча. Након сваке одигране игре бришу се резултати за сваког играча.
- Сценарио D. Игру чине по N агената сваке од доступних изведених класа класе *Player*. Након што свака два играча одиграју по један меч, X најгоре пласираних играча се избацује из игре, а за Y најбоље пласираних играча се додаје по један агент истог типа. Игра се понавља док год постоје барем два играча. Након сваке одигране игре бришу се резултати за сваког играча.
- Сценарио E. Игру чине по N агената сваке од доступних изведених класа класе *Player*. Након што свака два играча одиграју по један меч, X најгоре пласираних играча се избацује из игре и игра се понавља док год постоје барем два играча. Резултати за сваког играча се акумулирају након сваке одигране игре.
- Сценарио F. Игру чине по N агената сваке од доступних изведених класа класе *Player*. Након што свака два играча одиграју по један меч, X најгоре пласираних играча се избацује из игре и игра се понавља док год постоје барем два играча. Након сваке одигране игре бришу се резултати за сваког играча. При повлачењу потеза сваки играч може да направи грешку са вероватноћом M%.

Имати у виду да се на одбрани домаћег задатка могу појавити нови типови агената у игри. Имати у виду да се на одбрани домаћег задатка могу променити параметри наведених сценарија.

Имплементација се ради у програмском језику Јава уз дозвољено коришћење свих стандардних библиотека и структура података.

Класа студентског агента мора да се нађе у пакету `rs.ac.bg.etf.players`.



### Напомене

Електронску верзију решења овог домаћег задатка послати најкасније до 16. новембра 2020. године у 23:59h, као ZIP архиву на следећем [линку](#). Предаја домаћег задатка аутоматски представља и пријаву за исти. **Одјава након предаје није могућа.** Предајом домаћег задатка студент губи могућност израде једног од следећа два домаћа задатка. Одбрана домаћег задатка планирана је 20.11.2020. године.

Домаћи 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 домаћи задатак се може бранити само у за то предвиђеном термину. Домаћи задатак вреди максимално 10 поена (уз могућност додатних поена за радове који се посебно истакну).

Студентима се препоручује да детаљно анализирају доступне агенте и њихове стратегије кроз симулатор и измену параметара сценарија пре започињања израде решења.

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