



Први пројектни задатак из предмета Интелигентни системи школске 2022/23. године

Основне информације

Pytnik је графичка симулација написана на програмском језику Пајтон која приказује рад основних алгоритама претраживања на проблему трговачког путника. Главни прозор апликације приказује мапу на којој се налазе златници које агент треба да сакупи. Циљ агента је сакупљање свих златника и враћање у почетну позицију путањом коју изабрани алгоритам генерише.

Начин коришћења апликације

Програм се покреће из терминала навођењем следеће команде:

```
.\main.py map agent timeout
```

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 агента који се користи (подразумевано `ExampleAgent`)
- `timeout` – максимално време предвиђено за извршавање алгорита (подразумевано 5 секунди)

Пре покретања потребно је инсталирати пакет `pygame` (верз. 2.1.2) у оквиру Пајтон интерпретера. Након покретања приказује се главни прозор апликације:





У оквиру прозора приказана је мапа са нумерисаним златницима које агент треба да сакупи. Са десне стране налази се секција са информацијама о до тада обиђеним парцијалним путањама и тренутној цени путање. Симулација може да се покрене када се изабрани алгоритам изврши. Притиском на дугме *SPACE* могуће је покренути и привремено зауставити кретање агента. Притиском на дугме *ENTER* могуће је приказати резултат након сакупљања свих златника. Притиском на дугме *ESC* прекида се рад апликације и затвара се њен главни прозор. Притиском на дугме *S* апликација прелази у режим или излази из режима корак-по-корак и тада је могуће извршавати симулацију један корак унапред притиском на десну стрелицу или један корак уназад притиском на леву стрелицу.

Mana

Мапа је текстуална датотека следећег формата:

Свака линија у датотеци садржи информације о позицији једног златника. Појединачне информације у линији одвојене су зарезима. Прве две вредност одговарају x (у опсегу од 0 до 1000) и y (у опсегу од 0 до 600) координатама тачке на мапи у којој се исцртава златник. Координатни почетак (тачка $[0, 0]$) налази се у горњем левом углу апликације. Након координата наводи се одређен број вредности које представљају цене путева између текућег златника и (редом) свих златника до тада наведених у датотеци. Сматрати да је граф потпун и неусмерен па је пут од златника B до златника A једнак путу од златника A до златника B . Агент се при покретању апликације исцртава на месту првог наведеног златника.

Пример фајла мапе:

100,100

100,300,25

300,100,17,30

300,300,12,22,31





На основу улазне датотеке креира се матрица цена између свака два златника. Визуелни приказ матрице након учитавања датотеке из претходног примера дат је у наставку:

Идентификациона ознака златника	0	1	2	3
0	0	25	17	12
1	25	0	30	22
2	17	30	0	31
3	12	22	31	0



Агенти система

Потребно је имплементирати следеће агенте, сваког са својом стратегијом претраживања:

- Aki. Агент користи стратегију похлепне претраге по дубини тако што приликом избора наредног златника за сакупљање бира онај до ког је цена пута најмања. Уколико постоје два или више таквих златника, као следећи се бира онај са мањом вредношћу идентификационе ознаке. 
- Jocke. Агент користи *brute-force* стратегију тако што генерише све могуће путање и од свих бира ону са најмањом ценом. 
- Uki. Агент користи стратегију гранања и ограничавања. Уколико постоје две или више парцијалних путања истих цена, агент бира ону са више сакупљених златника на путањи, а у случају две или више таквих парцијалних путања бира ону која доводи до златника са мањом вредношћу идентификационе ознаке. 
- Micko. Агент користи A^* стратегију претраживања, при чему се за хеуристичку функцију користи минимално обухватно стабло. Уколико постоје две или више парцијалних путања истих вредности функције процене, агент бира ону са више сакупљених златника на путањи, а у случају две или више таквих парцијалних путања бира ону која доводи до златника са мањом вредношћу идентификационе ознаке. 

Кориснички захтеви

Студенти треба да имплементирају стратегије претраге проширивањем класе Agent (чија дефиниција треба да се нађе у датотеци `sprites.py`) и редефинисањем њене функције `get_agent_path`, чији параметар `coin_distance` представља квадратну матрицу цена путева између свака два златника, а повратна вредност је коначна путања. Она је представљена листом идентификационих ознака златника које чине путању од златника са идентификационом ознаком 0, до истог тог златника сакупљајући кроз све остале златнике на мапи. Студентима је у оквиру кода дат пример класе (ExampleAgent) која проширује основну класу Agent и на једноставан начин имплементира функцију `get_agent_path`.

На располагању су две мапе које студенти могу да користе за проверу свог решења. У наставку је дат пример покретања програма са датом пример класом:

- `.\main.py maps\map0.txt ExampleAgent 10`

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 16. новембра 2022. године до 20 часова, као ZIP архиву на следећем [линку](#). Одбрана домаћег задатка планирана је 18.11.2022. 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).

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`