

Вештачка интелигенција-2021/2022/L

Dashboard / My courses / Ви-2021/2022/L / Самостојна проверка на знаење / Тест 1

Quiz navigation

1 2

Finish attempt ...

Question 1

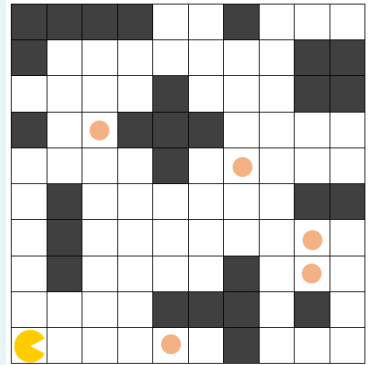
Not complete

Marked out of 10.00

Flag question

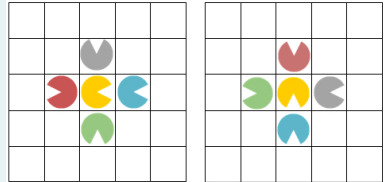
Предложете соодветна репрезентација на играта Расман и напишете ги потребните функции во Python за да се реши следниот проблем за кој една можна почетна состојба е прикажана на Слика 1:

Слика 1:



"Во табла со димензии 10x10 се наоѓа човечко. Човечето може да се придвижува на кое било соседно поле хоризонтално или вертикално, доколку на соодветната позиција не постои пречка. Потребно е човечето да ги изведе сите точки поставени во таблата. Во даден момент можни се четири акции на движење на човечето: продолжи право, продолжи назад, сврти лево и сврти десно. На Слика 2 се прикажани можните движења на човечето за две насоки, каде што со сина боја е обележана новата позиција добиена со акцијата продолжи право, продолжи назад со црвена боја, сврти лево со сива боја и сврти десно со зелена боја. Потребно е проблемот да се реши во најмал број на потези."

Слика 2:



За сите тест примери изгледот и големината на таблата се исти како на примерот даден на Слика 1. За сите тест примери позициите на пречките се исти. За секој тест пример почетната позиција на човечето се менува, а исто така се менуваат и позициите на точките.

Од стандарден влез се читаат почетните x и y координати во кои на почетокот се наоѓа човечето (ако таблата ја гледате во стандардниот координатен систем). Следно се чита насоката кон која е поставен играчот ('istok', 'zapad', 'sever', 'jug'). Потоа се чита број на точки во таблата, по што во секој нов ред се читаат x и y координатите на точките во таблата (ако таблата ја гледате во стандардниот координатен систем).

Движењата на човечето потребно е да ги именувате на следниот начин:

- **ProdolzhPravo** - за придвижување на човечето за едно поле напред
- **ProdolzhNazad** - за придвижување на човечето за едно поле на назад
- **SvrtiLevo** - за придвижување на човечето за едно поле налево
- **SvrtiDesno** - за придвижување на човечето за едно поле надесно

Вашиот код треба да има само еден повик на функција за приказ на стандарден излез (print) со кој ќе ја вратите секвенцата на движења која човечето треба да ја направи за да може од својата почетна позиција да стигне до позицијата на кукичката. Треба да примените неинформирано пребарување. Врз основа на тест примерите треба самите да определите кое пребарување ќе го користите.

**НАПОМЕНА:** Подреденоста на акциите во successor функција е важна кај неинформирано пребарување. Соодветно, за да се добие решението кое се очекува во изгенерираните излези, редоследот треба да биде ProdolzhPravo, ProdolzhNazad, SvrtiLevo, SvrtiDesno. Доколку акциите не се подредени со истиот редослед, можно е да се најде исто оптимално решение со различна патека.

For example:

Input	Result
0 0 istok 5 2,6 4,0 6,5 8,2 8,3	['ProdolzhPravo', 'ProdolzhPravo', 'ProdolzhPravo', 'ProdolzhPravo', 'ProdolzhNazad', 'ProdolzhPravo', 'SvrtiDesno', 'ProdolzhPravo', 'ProdolzhPravo', 'ProdolzhPravo']

Answer: (penalty regime: 0 %)

Reset answer

```
1 import bisect
2
3
4
5 Дефинирање на класа за структурата на проблемот кој ќе го решаваме со пребарување.
6 Класата Problem е апстрактна класа од која правиме наследување за дефинирање на основните
7 карактеристики на секој проблем што сакаме да го решиме
8
9
10
11 class Problem:
12     def __init__(self, initial, goal=None):
13         self.initial = initial
14         self.goal = goal
15
16     def successor(self, state):
17         """За дадена состојба, врати речник од парови {акција : состојба}
```

```
18     достапни од оваа состојба. ако има многу следовници, употребуете
19     итератор кој би ги генерирал следбениците еден по еден, наместо да
20     ги генерирате сите одеднаш.
21     :param state: дадена состојба
22     :return: речник од парови {акција : состојба} достапни од оваа
23             состојба
24     :rtype: dict
25     """
26     raise NotImplementedError
27
28     def actions(self, state):
29         """За дадена состојба state, врати листа од сите акции што може да
30         се применат над таа состојба
31         :param state: дадена состојба
```

Check

→ Announcements

Jump to...

Next page

You are logged in as [Велчевски Атанас](#) (Log out)

[Би:2021/2022/L](#)

[Data retention summary](#)

