

Trabalho individual prático nº 1

Programação ao estilo funcional em Python

Inteligência Artificial / Introdução à Inteligência Artificial
Ano Lectivo de 2015/2016

16 de Outubro de 2015

I Observações importantes

1. Este trabalho deverá ser entregue no prazo de 24 horas após a publicação deste enunciado. Os trabalhos poderão ser entregues para além do prazo, mas serão penalizados em 5% por cada hora adicional.
2. Submeta as funções pedidas num único ficheiro com o nome "tpi1.py" e inclua o seu nome e número mecanográfico. Casos de teste, instruções de impressão e código não relevante devem ser comentados ou removidos.
3. O trabalho está centrado na programação ao estilo funcional, por isso não deverá usar ciclos.
4. Pode discutir o enunciado com colegas, mas não pode copiar programas, ou partes de programas, qualquer que seja a sua origem.
5. Se discutir o trabalho com colegas, inclua um comentário com o nome e número mecanográfico desses colegas. Se recorrer a outras fontes, identifique essas fontes também.
6. Todo o código submetido deverá ser original; embora confiando que a maioria dos alunos fará isso, serão usadas ferramentas de detecção de copiar. Alunos que participem em casos de copiarão terão os seus trabalhos anulados.
7. Os programas serão avaliados tendo a conta: correcção e completude (70%); estilo (10%); e originalidade / evidência de trabalho independente (20%). A correcção e completude serão normalmente avaliadas através de teste automático. Se necessário, os módulos submetidos serão analisados pelos docentes para dar o devido crédito ao esforço feito.

II Exercícios

Quem tira férias fora dos meses de verão tipicamente não vai para a praia, dando preferência a passeios turísticos com visitas a vários locais de interesse. Frequentemente, estes passeios são

organizados pelas agências de viagens para grupos mais ou menos alargados. Como nada se faz nos dias de hoje sem suporte informático, as agências de viagens têm interesse em adquirir ferramentas que facilitem a gestão dos passeios turísticos, bem como extracção de diversa informação após a realização desses passeios.

No presente trabalho pretende-se que, partindo do registo de deslocações realizadas no âmbito de um passeio turístico, sejam extraídas diversas informações.

Para melhor compreensão, aqui fica um exemplo desse registo de deslocações:

```
>>> viagem = \
    [ ("aviao", "SaCarneiro", 10, "Stansted", 12),
      ("comboio", "Stansted", 13, "LiverpoolStreet", 14),
      ("metro", "LiverpoolStreet", 14, "WestBrompton", 15),
      ("metro", "WestBrompton", 16, "OxfordStreet", 17),
      ("metro", "OxfordStreet", 21, "WestBrompton", 22),
      ("metro", "WestBrompton", 9, "Victoria", 10),
      ("autocarro", "Victoria", 11, "Bristol", 14),
      ("autocarro", "Bristol", 10, "Victoria", 13),
      ("metro", "Victoria", 13, "Westminster", 14),
      ("metro", "Westminster", 19, "CoventGarden", 20),
      ("metro", "CoventGarden", 22, "WestBrompton", 23),
      ("metro", "WestBrompton", 9, "LiverpoolStreet", 10),
      ("metro", "LiverpoolStreet", 10, "Stansted", 11),
      ("aviao", "Stansted", 12, "SaCarneiro", 14)
    ]
>>>
```

Como pode ver, o passeio está descrito através de uma lista de deslocações. Cada deslocação está por sua vez descrita por um tuplo (T,O,HO,D,HD), em que T é o meio de transporte, O é o local de origem, HO é a hora aproximada de partida, D é o local de destino e HD é a hora aproximada de chegada.

A deslocações estão ordenadas por ordem cronológica. Pode assumir que, em cada deslocação, a hora de chegada é sempre maior que a hora de partida. Quando a hora de partida numa deslocação é menor que a hora de chegada na deslocação anterior, tal significa que a deslocação anterior ocorreu no dia imediatamente anterior.

Desenvolva agora as funções descritas em seguida (com os nomes indicados nos exemplos):

1. Uma função que, dado o registo de deslocações, devolva uma lista com todos os locais visitados, sem repetições. [Nota: Para tirar repetições neste exercício não poderá usar a conversão para set.]

Exemplo:

```
>>> locais(viagem)
['Stansted', 'SaCarneiro', 'LiverpoolStreet', 'WestBrompton',
 'CoventGarden', 'Westminster', 'Victoria', 'Bristol', 'OxfordStreet']
>>>
```

2. Uma função que, dado o registo de deslocações e um local específico, calcule o tempo médio de permanência nesse local considerando as várias passagens ou visitas feitas.

Exemplos:

```
>>> tempo_medio_em_loc(viagem, "WestBrompton")
7.33333333333
>>> tempo_medio_em_loc(viagem, "Westminster")
```

```

5.0
>>> tempo_medio_em_loc(viagem,"Victoria")
0.5
>>>

```

3. Uma função que, dado o registo de deslocações, determine o local central do passeio turístico, definido como sendo o local com maior número de passagens. Para este efeito, considere que não ocorrem situações de empate.

Exemplo:

```

>>> local_central(viagem)
WestBrompton
>>>

```

4. Uma função que, dado o registo de deslocações, determine as etapas principais do passeio. Essas etapas principais são sequências de deslocações que começam e/ou terminam no local central do passeio. Para cada etapa, deve também ser calculado o tempo total consumido nas deslocações.

Exemplo:

```

>>> etapas_principais(viagem)
[ (4, ['SaCarneiro', 'Stansted', 'LiverpoolStreet', 'WestBrompton'])
  (2, ['WestBrompton', 'OxfordStreet', 'WestBrompton'])
  (10, ['WestBrompton', 'Victoria', 'Bristol', 'Victoria',
        'Westminster', 'CoventGarden', 'WestBrompton'])
  (4, ['WestBrompton', 'LiverpoolStreet', 'Stansted', 'SaCarneiro']) ]
>>>

```

III Esclarecimento de dúvidas

Pedidos de esclarecimentos de dúvidas enviados aos docentes (lsl@ua.pt, dgomes@ua.pt) serão respondidos aqui. Bom trabalho!

1. Na pergunta 2 "médido" é "médio", certo?
Resposta: Certo. Já está corrigido acima.
2. Pode-se usar listas de compreensão?
Resposta: Sim.
3. Se usar um for numa lista de compreensão, o for é um ciclo?
Resposta: Uma lista de compreensão é uma expressão, e não um ciclo. Podem usar.
4. Há qualquer tipo de penalização pelo facto de se usar o método append numa lista?
Resposta: Não!
5. Podemos usar funções auxiliares?
Resposta: Certamente!

6. O registo de deslocações, caso não vazio, tem o formato que é suposto ter?

Resposta: Sim, podem assumir que as estruturas estão formadas como descrito no enunciado.

7. Até que matéria posso usar para resolver os problemas de IA?

Resposta: Toda a matéria do primeiro guião prático excluindo classes.

8. Não é possível usar dicionários, uma vez que ainda não foi leccionado certo?

Resposta: Certo.

9. No exercício 2 quando o tempo de chegada e de saída são iguais optamos por ter estado 0 horas ou 24 horas, no exemplo o professor opta por 0 horas fazemos o mesmo?

Resposta: Correcto. Quando a chegada e a saída seguinte tem a mesma hora, considera-se que é no mesmo dia.

10. A origem de cada deslocação é sempre o destino da anterior?

Resposta: Sim, assumam isso.

11. No exercício 2 para o caso de SaCarneiro o resultado seria zero ou terá que dar erro por não conseguirmos calcular?

Resposta: Nesse caso, uma vez que não há hora de chegada e partida, faz mais sentido dar None.

12. Na alínea 2, podemos considerar que, durante o passeio, não volta lá pelo meio à posição de origem?

Resposta: Esse caso especial não trás problemas e por isso não há razão para o excluir.