



Introdução à Programação e Resolução de Problemas

2015/2016

Teste Final

7 de janeiro de 2016 - 9h00, Duração: 2h30

Nome:

Número:

Turma:

AVISO

O Teste Final tem a duração de 2h30 min e é composto por cinco perguntas a que deverá responder sem recurso ao computador. As respostas às perguntas devem ser dadas no espaço reservado para o efeito. Como material de consulta pode usar apenas o manual de referência do Python previamente anunciado na inforestudante. A violação desta última regra implica a anulação da prova, reprovação à cadeira e eventual processo disciplinar. Caso tenha alguma dúvida pergunte. Boa sorte!

Pergunta	Pontos	Resultado
1)	20	
2)	20	
3)	20	
4)	20	
5)	20	
	100	

Visto por:

Pergunta 1 **20 pontos**

- a) Diga o que entende por **parâmetros formais** e **parâmetros reais**.
Mostre, através de código exemplo, como se ligam.

A sua resposta:

- b) Considere a seguinte sessão no interpretador. Indique o que vai aparecer no lugar dos pontos de interrogação, e explique com rigor a razão para a sua resposta, não se esquecendo de usar as linhas como referência.

```
1 >>> def func(l1, l2):
2 ...     for elemento in l2:
3 ...         l1.append(elemento)
4 ...
5 >>> lista1 = [1, 2, 3]
6 >>> lista2 = [1, 2, 3]
7 >>> func(lista1, lista2)
8 >>> print(func(lista1, lista2))
9 ??
10 >>> lista1
11 ??
```

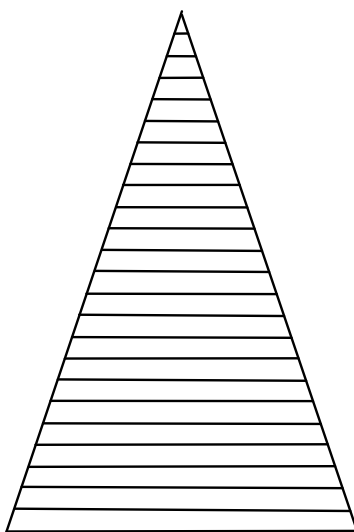
A sua resposta:

Nome:

Número:

Pergunta 2 **20 pontos**

Quando se olha para uma linha de caminho de ferro numa reta que seja longa, os carris parecem unir-se num ponto distante. Nesta pergunta, deverá usar o módulo `turtle` para desenhar uma linha de caminho de ferro em perspetiva, semelhante à que está representada na figura seguinte:



Deverá permitir que o comprimento dos lados, da base e o número de travessas sejam configuráveis. Serão valorizadas soluções modulares.

A sua resposta:

A sua resposta:

Nome:

Número:

Pergunta 3 **20 pontos**

Implemente uma versão sua do método das cadeias de caracteres count, que retorna o número de ocorrências não sobrepostas de uma subcadeia numa cadeia de caracteres. Por exemplo:

a subcadeia "lal" aparece apenas uma vez na cadeia "olalal mundo".

Não pode recorrer a métodos das strings, com exceção do método len para calcular o tamanho da mesma.

A sua resposta:

A sua resposta:

Nome:

Número:

Pergunta 4 20 pontos

Existem vectores em que a grande maioria das suas componentes vale zero. Esses vectores chamam-se esparsos. Uma forma simples de os representar em Python é baseada em dicionários: cada chave é a posição no vector e o valor é um número. Por exemplo, o vector $x_1 = (0, 0, 0, 5, 0, 0, 2)$ pode ser representado pelo dicionário $d = \{3:5, 6:2, \text{'len':}7\}$. A soma de dois vectores, esparsos ou não, é dada pela soma dos respectivos elementos. Por exemplo, com $x_2 = (0, 2, 0, 4, 0, 3, 0)$, temos que $x_1 + x_2 = (0, 2, 0, 9, 0, 3, 2)$. Escreva um programa que dados dois vectores esparsos representados por dicionários devolva o dicionário que representa a sua soma. Pode assumir que os dois vectores têm a mesma dimensão.

A sua resposta:

A sua resposta:

Nome:

Número:

Pergunta 5 20 pontos

Considere que tem num ficheiro informação sobre uma série de vectores. Em cada linha tem representado um vector, com as suas componentes separadas por vírgula. Por exemplo:

```
0,0,0,2,0,3,0
0,0,0,0,0,2,0
0,0,0,0,1,0,0
0,0,0,5,0,0,0
```

Escreva um programa que, dado o nome de um ficheiro com esta informação, devolve o vector soma de todos os vectores no ficheiro, representado através de um dicionário semelhante ao indicado na pergunta anterior. Pode utilizar, se assim o entender, a função criada na pergunta anterior.

A sua resposta:

A sua resposta: