

Quiz navigation



Finish review

Started on	Monday, 23 November 2020, 8:59 AM
State	Finished
Completed on	Sunday, 29 November 2020, 10:59 PM
Time taken	6 days 13 hours
Grade	12.00 out of 12.00 (100%)

Question 1

Correct

Mark 2.00 out of 2.00

Flag question

Defina a função **contarDigitos** que recebe um inteiro  $n$  e devolve o número de dígitos que contém. A função não deve ser recursiva, devendo resolver o exercício com o uso de ciclos.

For example:

Test	Result
<code>print(contarDigitos(123))</code>	3
<code>print(contarDigitos(-1000))</code>	4

Answer: (penalty regime: 0 %)

Reset answer

```
1 def contarDigitos(n):
2     ## TODO
3     string=(str(n))
4     if n==0:
5         return 1
6     while n<1:
7         elementos=len(string)
8         return elementos -1
9     return len(string)
```

	Test	Expected	Got	
✓	<code>print(contarDigitos(123))</code>	3	3	✓
✓	<code>print(contarDigitos(-1000))</code>	4	4	✓

Passed all tests! ✓

Correct

Marks for this submission: 2.00/2.00.

Question 2

Correct

Mark 2.00 out of 2.00

Flag question

Defina a função **idxMaxPar** que recebe uma lista de inteiros positivos e devolve o índice onde se encontra o maior número par da lista. Se a lista não contiver números pares, a função deve devolver -1.

For example:

Test	Result
<code>print(idxMaxPar([3, 7, 2, 1, 7, 9, 10, 13]))</code>	6
<code>print(idxMaxPar([1, 3, 5, 7]))</code>	-1

Answer: (penalty regime: 0 %)

Reset answer

```
1 def idxMaxPar(x):
2     if all(i % 2 != 0 for i in x):
3         return -1
4     par = [i for i in x
5            if i % 2 == 0]
6     maior=max(par)
7     return x.index(maior)
```

	Test	Expected	Got	
✓	<code>print(idxMaxPar([3, 7, 2, 1, 7, 9, 10, 13]))</code>	6	6	✓
✓	<code>print(idxMaxPar([1, 3, 5, 7]))</code>	-1	-1	✓

Passed all tests! ✓

Correct

Marks for this submission: 2.00/2.00.

Question 3

Correct

Mark 2.00 out of 2.00

Flag question

Defina a função **neutralizarSinais** que recebe duas *strings*  $s1$  e  $s2$  de igual tamanho compostas por símbolos '+'-. A função deve devolver uma *string* do mesmo tamanho onde o  $i$ -ésimo caracter deve ser:

- '+' se  $s1[i]$  e  $s2[i]$  forem ambas '+'
- '-' se  $s1[i]$  e  $s2[i]$  forem ambas '-'
- '0' caso contrário (ou seja, os sinais opostos neutralizam-se)

For example:

Test	Result
<code>print(neutralizarSinais("++", "--"))</code>	<code>+-0</code>
<code>print(neutralizarSinais("---+--", "++---+"))</code>	<code>000000</code>

Answer: (penalty regime: 0 %)

Reset answer

```
1 def neutralizarSinais(s1, s2):
2     if len(s1)==0:
3         return ''
4     else:
5         ss1=[x for x in s1]
6         ss2=[x for x in s2]
7         resultados=[]
8         for i in range(len(ss1)):
9             for x in ss1[i]:
10                for y in ss2[i]:
11                    if x==y:
12                        resultados.append(s1[i])
13                    else:
14                        resultados.append('0')
15     return resultados2(resultados)
```

	Test	Expected	Got	
✓	<code>print(neutralizarSinais("++", "--"))</code>	<code>+-0</code>	<code>+-0</code>	✓
✓	<code>print(neutralizarSinais("---+--", "++---+"))</code>	<code>000000</code>	<code>000000</code>	✓

Passed all tests! ✓

Correct

Marks for this submission: 2.00/2.00.

Question 4

Correct

Mark 2.00 out of 2.00

Flag question

Defina a função **somaAcc** que recebe uma lista de inteiros e devolve a lista com as somas acumuladas de ir juntando mais um elemento de cada vez, e pela ordem inicial dada.

Por exemplo, **somaAcc(1,2,3,4)** deve retornar a lista  $[1, 1+2, 1+2+3, 1+2+3+4]=[1, 3, 6, 10]$ .

For example:

Test	Result
<code>print(somaAcc([1,2,3,4]))</code>	<code>[1, 3, 6, 10]</code>
<code>print(somaAcc([]))</code>	<code>[]</code>
<code>print(somaAcc([4,3,2,1]))</code>	<code>[4, 7, 9, 10]</code>

Answer: (penalty regime: 0 %)

Reset answer

```
1 def somaAcc(lista):
2     if lista==[]:
3         return []
4     for i in range(1, len(lista)):
5         lista[i]=lista[i] + lista[i-1]
6     return lista
```

	Test	Expected	Got	
✓	<code>print(somaAcc([1,2,3,4]))</code>	<code>[1, 3, 6, 10]</code>	<code>[1, 3, 6, 10]</code>	✓
✓	<code>print(somaAcc([]))</code>	<code>[]</code>	<code>[]</code>	✓
✓	<code>print(somaAcc([4,3,2,1]))</code>	<code>[4, 7, 9, 10]</code>	<code>[4, 7, 9, 10]</code>	✓

Passed all tests! ✓

Correct

Marks for this submission: 2.00/2.00.

Question 5

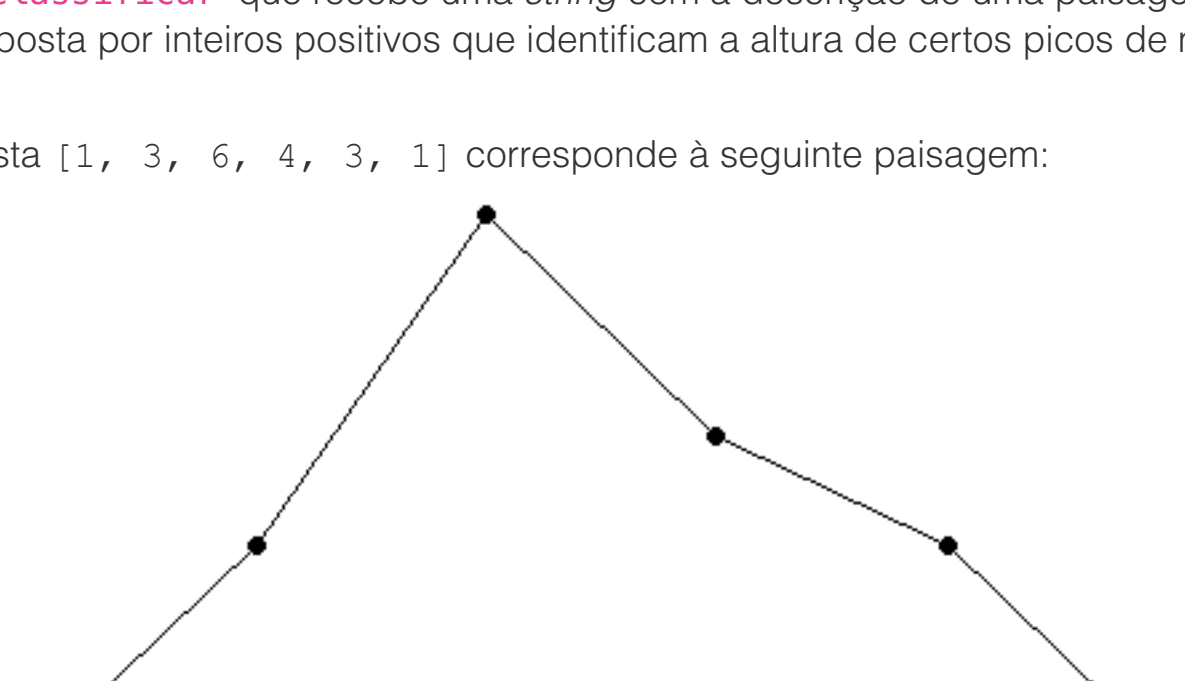
Partially correct

Mark 4.00 out of 4.00

Flag question

Defina a função **classificar** que recebe uma *string* com a descrição de uma paisagem. Esta paisagem é composta por inteiros positivos que identificam a altura de certos picos de montanha ou vales.

Por exemplo, a lista  $[1, 3, 6, 4, 3, 1]$  corresponde à seguinte paisagem:



O objetivo é classificar uma dada paisagem, a função deve devolver a *string* de valor:

- 'montanha' se houver apenas um pico (o pico não pode ocorrer numa das extremidades)
- 'vale' se houver apenas uma depressão (a depressão não pode ocorrer numa das extremidades)
- 'nenhum' se a paisagem não for um dos casos acima

No caso da lista  $[1, 3, 6, 4, 3, 1]$  o resultado esperado é a *string* 'montanha'

For example:

Test	Result
<code>print( classificar([3, 4, 5, 4, 3]) )</code>	montanha
<code>print( classificar([9, 7, 3, 1, 2, 4]) )</code>	vale
<code>print( classificar([9, 7, 9, 4]) )</code>	nenhum

Answer: (penalty regime: 0 %)

Reset answer

```
1 def classificar(paisagem):
2     maximo=max(paisagem)
3     indiceMax=paisagem.index(maximo)
4     indiceSeguinteMax=paisagem.index(maximo)+1
5     if paisagem[indiceSeguinteMax]==paisagem[indiceMax]:
6         return 'nenhum'
7     else:
8         if len(paisagem)<=3:
9             return 'nenhum'
10            p = paisagem.index(max(paisagem))
11            for i in range (0,p):
12                (paisagem[i] < paisagem[i+1])
13            for q in range (p,len(paisagem)-1):
14                (paisagem[q]> paisagem[q+1])
15            return 'montanha'
```

	Test	Expected	Got	
✓	<code>print( classificar([3, 4, 5, 4, 3]) )</code>	montanha	montanha	✓
✓	<code>print( classificar([9, 7, 3, 1, 2, 4]) )</code>	vale	vale	✓
✓	<code>print( classificar([9, 7, 9, 4]) )</code>	nenhum	nenhum	✓

Your code failed one or more hidden tests.

Partially correct

Marks for this submission: 2.86/4.00.

Finish review