

Tuplas em Python

INF 1026 – Computação Aplicada

Pontifícia Universidade Católica
Departamento de Informática

Revisão: Exercício

Faça um programa capaz de corrigir uma prova de múltipla escolha realizada por uma turma de alunos.

Como a prova só está disponível em um computador, os alunos respondem esta prova online, em ordem de chamada.

Portanto, quando o aluno entra na sala de provas, fornece sua matrícula e responde a prova. Ao término de sua avaliação o programa mostra sua nota.

A aplicação da prova é encerrada quando um código de matrícula 0 é fornecido.

A prova tem 10 questões, valendo 1 ponto cada uma.

Cada questão tem 5 alternativas ('a' até 'e').

Revisão: Corrigir prova de múltipla escolha

Questão 1

Os sobrinhos do personagem da Disney chamado de Pato Donald são:

- a) Huguinho, Zézinho e Paulinho.
- b) Joãozinho, Zézinho, Huguinho e Paulinho.
- c) Juninho, Zézinho e Huguinho.
- d) Luizinho, Huguinho e Zézinho.
- e) Patinho, Patola e Patinhozinho.

Questão 2

A metade da metade de 12 é igual a:

- a) 6.
- b) 4.
- c) 2.
- d) 3.
- e) 12.

Questão 3

Nome do rio onde fica a Cachoeira de Paulo Afonso.

- a) São Francisco.
- b) Paraná.
- c) Tietê.
- d) Paraíba do Sul.
- e) Amazonas.

Questão 4

Uma dessas afirmações está ERRADA.

- a) A árvore símbolo que deu nome ao nosso país é o Pau Brasil.
- b) O Jacaré é um réptil.
- c) O Manute era um anfíbio.
- d) A lua é o satélite natural da terra.
- e) A piranha é um peixe de água doce.

Questão 5

Ao entrar numa sala, João contou 4 pessoas, incluindo ele. Todos estavam calçados. Sem contar com ele quantos sapatos havia na sala?

- a) 4.
- b) 6.
- c) 8.
- d) 16.
- e) 10.

Questão 6

Complete a frase a seguir: "De Grão em grão..."

- a) A galinha fica mais gorda.
- b) O galo fica maior.
- c) O pintinho vai ficando gordo.
- d) O Depósito fica cheio.
- e) A galinha enche o papo.

Questão 7

O animal já extinto chamado DODÔ, era:

- a) Um réptil.
- b) Um dinossauro.
- c) Um pássaro.
- d) Um peixe que media até 3 metros de comprimento.
- e) Uma serpente marinha que se alimentava exclusivamente de algas.

Questão 8

A palavra MARAJÁ quer dizer:

- a) Pessoa muito rica.
- b) Pessoa que vive sem fazer nada.
- c) Pessoa que ganha dinheiro sem trabalhar.
- d) Título de nobreza indiano.
- e) Espécie de gato silvestre selvagem.

Questão 9

A palavra MARACUTAIA tem quantas vogais:

- a) 4.
- b) 5.
- c) 6.
- d) 7.
- e) 8.

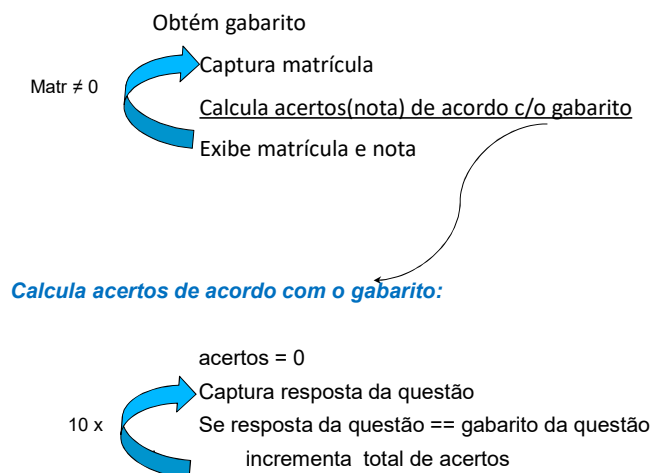
Questão 10

O Carnaval é:

- a) 38 dias antes da Páscoa.
- b) 39 dias antes da Páscoa.
- c) 40 dias antes da Páscoa.
- d) 42 dias antes da Páscoa.
- e) 45 dias antes da Páscoa.

3

Desenvolvendo a Solução

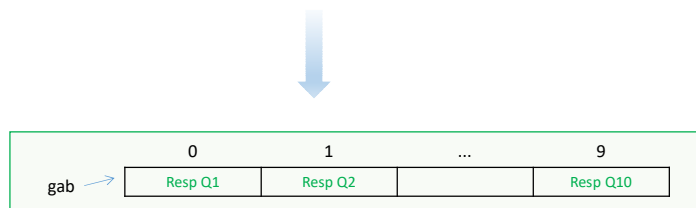


4

Armazenando o gabarito

Gabarito: 'd', 'd', 'a', 'd', 'b', 'e', 'c', 'c', 'c', 'c'

Sequência de respostas corretas



5

Revisão: Corrigir prova - Principal

Matr
≠
0

1) Obter o gabarito (sequência de respostas corretas)

2) Corrigir prova:

Obter respostas, contando acertos de acordo com o gabarito

Questão 1
Os sobrinhos do personagem da Disney chamado de Pató Donald são:
a) Huguiño, Zélinho e Paulinho.
b) Julezinho, Zélinho, Huguiño e Paulinho.
c) Juninho, Zélinho e Huguiño.
d) Lulinha, Huguiño e Zélinho.
e) Patrícia, Pató e Patríciozinho.

Questão 2
A metade da metade de 12 é igual a:
a) 6.
b) 4.
c) 2.
d) 1.
e) 12.

Questão 3
Nome do rio onde fica a Cachoeira de Paulo Afonso.
a) São Francisco.
b) Paraíba.
c) Tietê.
d) Paraíba do Sul.
e) Amazonas.

Questão 4
Uma das afirmações está ERRADA.
a) A árvore símbolo que deu nome ao nosso país é o Pau Brasil.
b) O Jacaré é um réptil.
c) O Hamule era um animal.
d) A lua é o satélite natural da terra.
e) A pirâmide é um peixe de água doce.

Questão 5
Ao entrar numa sala, João contou 4 pessoas, incluindo ele. Todos estavam calçados. Sem contar com ele quantos sapatos havia na sala?
a) 4.
b) 6.
c) 8.
d) 16.
e) 10.

Questão 6
Complete a frase a seguir: "De Grão em grão..."
a) A galinha fica mais gorda.
b) O galo fica mais.
c) O pintinho vai ficando gordo.
d) O Donatello fica cheio.
e) A galinha enche o papo.

Questão 7
O animal já extinto chamado DODO, era:
a) Um réptil.
b) Um dinossauro.
c) Um pássaro.
d) Um peixe que media até 3 metros de comprimento.
e) Uma serpente marinha que se alimentava exclusivamente de algas.


Questão 8
A palavra MABIA quer dizer:
a) Pessoa muito rica.
b) Pessoa que vive sem fazer nada.
c) Pessoa que ganha dinheiro sem trabalhar.
d) Título de nobreza indiano.
e) Espécie de gato selvagem.

Questão 9
A palavra MARACUTIA tem quantas vogais:
a) 5.
b) 6.
c) 7.
d) 8.
e) 9.

Questão 10
O Carnaval é:
a) 30 dias antes da Páscoa.
b) 39 dias antes da Páscoa.
c) 40 dias antes da Páscoa.
d) 41 dias antes da Páscoa.
e) 45 dias antes da Páscoa.

6


Revisão: Corrigir prova - Principal

- Matr
≠
0
- 
- 1) Obter o gabarito (sequência de respostas corretas)
 - 2) Corrigir prova:
Obter respostas, contando acertos de acordo com o gabarito

```
gab = ['d','d','a','d','b','e','c','c','c','c']
```

7

Revisão: Corrigir prova

- Matr
≠
0
- 
- 1) Obter o gabarito (sequência de respostas corretas)
 - 2) Corrigir prova:
Obter respostas, contando acertos de acordo com o gabarito

```
def corrige(gabarito):  
    nota = 0  
    for i in range(10):  
        resp = input("Digite a resposta da questão %d"%(i+1))  
        if resp == gabarito[i]:  
            nota+=1  
    return nota
```

8

Revisão: Corrigir provas da Turma

Matr
≠
0



1) Obter o gabarito (sequência de respostas corretas)

2) Corrigir prova:

Obter respostas, contando acertos de acordo com o gabarito

```
gab = ['d', 'd', 'a', 'd', 'b', 'e', 'c', 'c', 'c', 'c']
matr = int(input("Sua matrícula? "))
while matr != 0:
    nota = corrige(gab)
    print("%d - sua nota é: %d\n" % (matr, nota))
    matr = int(input("Sua matrícula? "))
```

9

Revisão: Corrigir prova – Nova versão

Alterar o programa para que o aluno obtenha a nota e uma lista com valores lógicos indicando as questões que acertou e as que errou.

Matr ≠ 0



Obtém gabarito

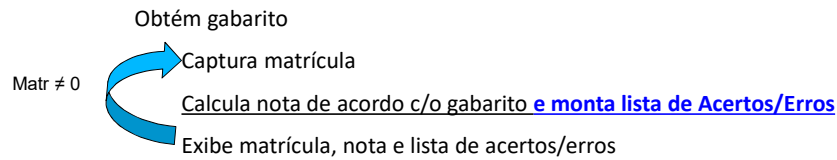
Captura matrícula

Calcula nota de acordo c/o gabarito e monta lista de Acertos/Erros

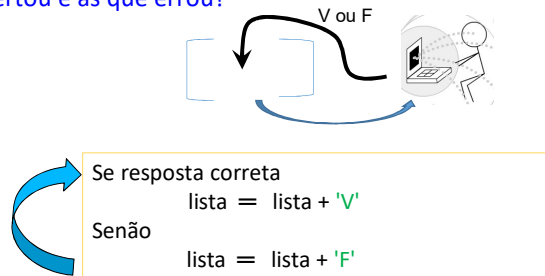
Exibe matrícula, nota e lista de acertos/erros

10

Desenvolvendo a Solução

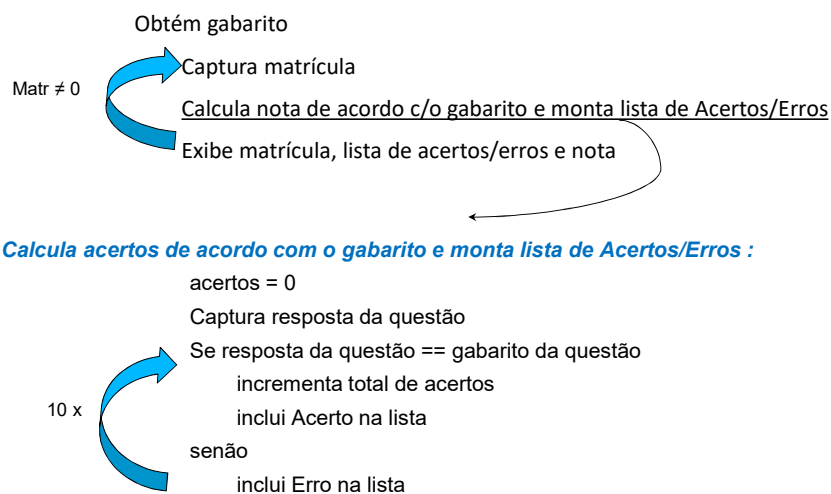


Como preencher uma lista com valores lógicos indicando as questões que o aluno acertou e as que errou?



11

Desenvolvendo a Solução



12

Revisão: Corrigir prova – Nova versão

Alterar o programa para que o aluno obtenha a nota e uma lista com valores lógicos indicando as questões que acertou e as que errou.

```
gab = ['d','d','a','d','b','e','c','c','c','c']
matr = int(input("Sua matrícula? "))
while matr != 0:
    lCorrecao = []
    nota = corrige(gab, lCorrecao)
    exibeCorrecao(matr,nota,lCorrecao)
    matr = int(input("Sua matrícula? "))
```

13

Revisão: Corrigir prova

```
def corrige(gabarito, lista):
    nota = 0
    for i in range(10):
        resp = input("Digite a resposta da questão %d: "%(i+1))
        if resp == gabarito[i]:
            nota+=1
            lista.append(True)
        else:
            lista.append(False)
    return nota

def exibeCorrecao(matr,nota,Lista):
    print("%d- Sua nota é: %d\n"%(matr,nota))
    print(Lista)
    return
```

14

Para casa

Alterar o programa para que a função **corrige** exiba o enunciado das questões a cada aluno. O enunciado das questões está no arquivo "questões.txt" e tem como separador o caractere '#'. O arquivo será disponibilizado.

Criar um arquivo texto, separados por ';' a partir do Excel, com o gabarito. Alterar o programa, criando a função **carrega** que deverá ler este arquivo na variável gabarito.

15

Revisão: Corrigir prova Analisando a solução

```
def corrige(gabarito, lista):  
    nota = 0  
    for i in range(10):  
        resp = input("Digite a resposta da questão %d: "%(i+1))  
        if resp == gabarito[i]:  
            nota+=1  
            lista.append(True)  
        else:  
            lista.append(False)  
    return nota  
  
def exibeCorrecao(nota, Lista):  
    print("Sua nota é: %d\n"%nota)  
    print(Lista)  
  
gab = ['d', 'd', 'a', 'd', 'b', 'e', 'c', 'c', 'c', 'c']  
matr = int(input("Sua matrícula? "))  
while matr != 0:  
    lCorrecao = []  
    nota = corrige(gab, lCorrecao)  
    exibeCorrecao(matr, nota, lCorrecao)  
    matr = int(input("Sua matrícula? "))
```

Por que usar
uma lista?
O gabarito é
imutável!!

Será que (e como)
uma função pode
retornar mais que um
valor??????

16

Tipo Tupla em Python

Tupla: uma sequência IMUTÁVEL de valores de qualquer tipo.

Valores de uma tupla: podem ser mutáveis e de tipos diferentes. São delimitados por parênteses (opcionais) e separados por vírgulas.

Exemplo: tuplas com gabaritos de provas com 5 questões:

```
gabMultiplaEscolha = ('a','b','c','a','d')
gabDiscursiva = (('MAO','MOA'), 13.8, 'c', 6, [2,3])
```

17

Tuplas são sequências

string: sequência imutável de caracteres

tupla: sequência imutável de elementos de qualquer tipo

lista: sequência mutável de elementos de qualquer tipo

```
Lista = [1, 2, 3, 4]
```

```
Tupla = (1, 2, 3, 4)
```

```
String = "12345"
```

Operações válidas em sequências:

- ✓ comprimento
- ✓ indexação
- ✓ concatenação
- ✓ replicação
- ✓ pertinência
- ✓ fatiamento

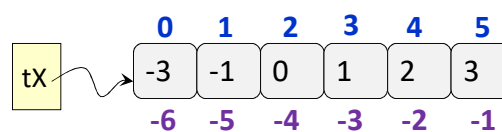
18

Tipo Tupla em Python - Índices

Os itens de uma tupla são identificados por **índices**, que variam de:

- 0 até comprimento da tupla-1 (posição a partir do início)
- -1 até -comprimento da tupla (posição a partir do final)

Exemplo: `tX = (-3,-1,0,1,2,3)`



19

Criando uma tupla por enumeração

Tupla vazia:

`nomeVariávelTupla = ()`

A vírgula é obrigatória

Tupla com um elemento :

`nomeVariávelTupla = (el1 ,)`

Tupla com elementos enumerados :

`nomeVariávelTupla = (el1 , el2 , ... , eln)`

20

Criando uma tupla por enumeração

Exemplos:

```
amigos = ( 'João', 'Pedro', 'Ana' )  
megaSena = (1,17,22,24,25,36)  
vazia = ()  
mista= ( 'Pat', 9.0, 15*2, [1,2, 'ze'], (2,2) )  
tuplas = (amigos, (1,2,3))  
tupla1El= ('1',)
```

Sem a vírgula, o Python entende ('1') como um valor entre ()

21

Comprimento de uma tupla

len (varTupla)

Retorna o número de elementos de uma tupla

Uma lista/tupla aninhada é UM elemento da tupla que a contém.

Exemplos:

```
amigos = ( 'João', 'Pedro', 'Ana' )  
tuplas = ( amigos, (1,2,3) )  
vazia = ()  
megaSena = ( 1,17,22,24,25,36 )  
mista = ( 'Pat', 9.0, 15*2, [1,2, 'ze'] )
```

```
len(amigos)    → 3  
len(tuplas)    → 2  
len(vazia)     → 0  
len(megaSena)  → 6  
len(mista)     → 4
```

22

Exibindo uma variável do tipo tupla

```
print (varTupla)
```

Exemplos:

```
amigos = ( 'João', 'Pedro', 'Ana' )  
megaSena = (1,17,22,24,25,36 )  
mista = ( 'Pat', 9.0, 15*2, [1,2,'ze'] )  
tuplas = ( amigos,megaSena )  
vazia = ()
```

```
print( 'Amigos: ', amigos)  
print( 'MegaSena: ', megaSena)  
print( 'Com subtupla: ', mista)  
print( 'Com tuplas: ', tuplas)  
print( 'Vazia: ', vazia)
```

```
Amigos: ('João', 'Pedro', 'Ana')  
MegaSena: (1, 17, 22, 24, 25, 36)  
Com subtupla: ('Pat', 9.0, 30, [1, 2, 'ze'])  
Com tuplas: (('João', 'Pedro', 'Ana'), (1, 17, 22, 24, 25, 36))  
Vazia: ()
```

23

Acesso aos elementos da tupla: Indexação

Pelo operador de indexação **[]**.

```
varTupla [expressão inteira]
```

expressão inteira: resultando em um valor inteiro e determina o índice.

- ✓ Índices positivos (0 até comprimento da tupla-1) indicam a posição do elemento na tupla da esquerda para a direita.
- ✓ Índices negativos (-1 até -comprimento da tupla) indicam a posição do elemento da direita para a esquerda.

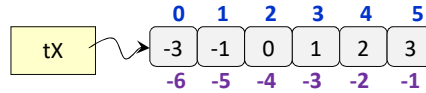
Índices fora do intervalo provocam um erro

24

Acesso aos elementos da tupla: Indexação

Exemplo 1:

tX = (-3,-1,0,1,2,3)



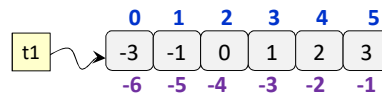
```
tam = len(tX)
print(tX[2])      → 0
print(tX[9-8])    → -1
print(tX[-2])     → 2
print(tX[tam-1])  → 3
print(tX[-1])     → 3
print(tX[tam])    → IndexError: tuple index out of range
```

25

Tupla é imutável

Exemplo 1:

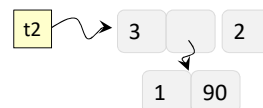
t1 = (-3,-1,0,1,2,3)



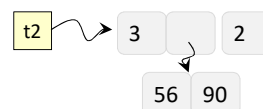
t1[0] = -2 → **TypeError: object doesn't support item assignment**

Exemplo 2:

t2 = (3,[1,90],2)



t2[1][0] = 56 → (3,[56,90],2)



26

Acesso aos elementos da tupla: Indexação

Exemplo 2:

```
vencConc = (('Jo',9),('Ana',8.9),('Zé',8.4))
```

```
print(vencConc[0])           → ('Jo', 9)
print(vencConc[-len(vencConc)]) → ('Jo', 9)
print(vencConc[-4])          → IndexError: tuple index out of range
print(vencConc[-1][0])       → 'Zé'
print(vencConc[-1][2])       → IndexError: tuple index out of range
```

27

Iterando sobre os elementos da tupla

for elemento in tupla:
CORPO

Acessa sequencialmente os elementos da tupla, do 1º ao último

Exemplo:

```
tupla = (1, 2, 3, 4)
for el in tupla:
    print(el)
```

```
Python 3.5.2 Shell
File Edit Shell Debug Options Window Help
Python 3.5.2 (tags/v3.5.2:9d78a20, Dec 23 2016, 22:01:18) [AMD64] on win32
Type "help()" or "license()" for more >>>
1
2
3
4
```

28

Iterando sobre os elementos da tupla

```
for indice in range(len(tupla)):
    CORPO
```

Gera os índices da tupla sequencialmente, de 0 a nº de elementos-1

Exemplo:

```
tupla = (1,2,3,4)
tam = len(tupla)
for i in range(tam):
    print(tupla[i])
```

```
Python 3.5.2 Shell
File Edit Shell Debug Options Window Help
Python 3.5.2 (tags/v3.5.2:90f62910a, Nov 25 2016, 22:01:18) [AMD64] on win32
Type "copyright", "credits" or "license()" for more information.

1
2
3
4
```

29

Iterando sobre os elementos da tupla

```
for indice, elem in enumerate(tupla):
    CORPO
```

Gera os índices da tupla sequencialmente, de 0 a nº de elementos-1 e
acessa o elemento do índice

Exemplo:

```
tupla = (1,2,3,4)
for i,el in enumerate(tupla):
    print(el, 'está na pos ', i)
```

```
Python 3.5.2 Shell
File Edit Shell Debug Options Window Help
Python 3.5.2 (tags/v3.5.2:90f62910a, Nov 25 2016, 22:01:18) [AMD64] on win32
Type "copyright", "credits" or "license()" for more information.

1 está na pos 0
2 está na pos 1
3 está na pos 2
4 está na pos 3
```

30

Concatenação: +

tupla1+tupla2

Cria uma nova tupla com os elementos da tupla1 seguidos dos elementos da tupla2

Exemplo:

```
T1 = (1, 2, 3, 4)
T2 = (-2, -1, 0, 1, 2, 3, 4, 5)
T3 = T1 + T2
print(T3)
```

(1, 2, 3, 4, -2, -1, 0, 1, 2, 3, 4, 5)

31

Replicação: *

tupla * n

cria uma nova tupla com *n* cópias de *tupla*, concatenadas

Exemplos:

```
T1 = (1, 2, 3, 4)
T2 = 3 * T1
print(T2)
```

(1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4)

32

Pertinência: in

el in tupla

retorna **True**, se *el* \in à *tupla* e **False**, caso contrário

el not in tupla

retorna **True**, se *el* \notin à *tupla* e **False**, caso contrário

Exemplos:

T1 = (1, 2, 3, 4)		T2 = ((2,3), (1,2))	
print(3 in T1)	True	print(3 in T2)	False
print(3 not in T1)	False	print(3 not in T2)	True
print((1,2) in T1)	False	print((1,2) in T2)	True
print((1,2) not in T1)	True	Print((1,2) not in T2)	False

33

Fatiamento: [a:b:n]

tupla[a:b:n]

cria uma nova tupla com os elementos da fatia selecionada.

Fatia iniciada em a (inclusive) até b (exclusive) de n em n

Exemplos:

```
print(t[2:5])
```

```
(0, 1, 2)
```

```
print(t[-6:-3])
```

```
(-2, -1, 0)
```

```
print(t[-1::-1])
```

```
(3, 2, 1, 0, -1, -2)
```

t = (-2, -1, 0, 1, 2, 3)



34

Fatiamento

- tupla[a : b]** - cria uma cópia de **a** (**inclusive**) até **b** (**exclusive**)
- tupla[a :]** - cria uma cópia a partir de **a** (**inclusive**)
- tupla[: b]** - cria uma cópia até **b** (**exclusive**)
- tupla[:]** - cria uma cópia de todos os elementos
- tupla[a : b : n]** - cria uma cópia de **a** (**inclusive**) até **b** (**exclusive**) de **n** em **n** elementos

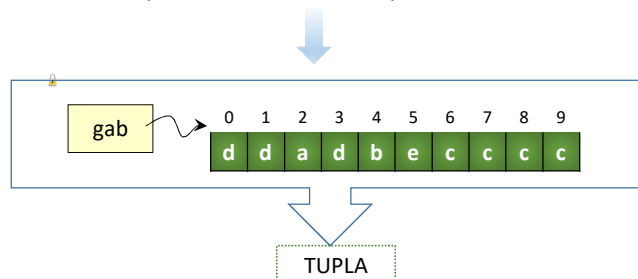
35

Voltando ao Exercício

Que valores foram armazenados inapropriadamente em listas?

Gabarito: 'd', 'd', 'a', 'd', 'b', 'e', 'c', 'c', 'c', 'c'

Sequência IMUTÁVEL de respostas corretas



Refazer o exercício usando a estrutura mais adequada.

36

Corrigir prova – outra solução

Matr
≠
0

1) Ler o gabarito do arquivo numa tupla

2) Corrigir prova:

Obter respostas, contando acertos de acordo com o gabarito

Questão 1
Os sobrinhos do personagem da Disney chamado de Pato Donald são:
a) Huguiinho, Zizinho e Paulinho.
b) Solzeinho, Zizinho, Huguiinho e Paulinho.
c) Juriinho, Zizinho e Huguiinho.
d) Lucioinho, Huguiinho e Zizinho.
e) Patinho, Patola e Patinhozinho.

Questão 2
A metade da metade de 12 é igual a:
a) 6.
b) 4.
c) 2.
d) 5.
e) 12.

Questão 3
Nome do rio onde fica a Cachoeira de Paulo Afonso.
a) São Francisco.
b) Paraíba.
c) Tietê.
d) Paraíba do Sul.
e) Amazonas.

Questão 4
Uma dessas afirmações está ERRADA.
a) A árvore símbolo que deu nome ao nosso país é o Pau Brasil.
b) O Jacaré é um réptil.
c) O Hemure era um anfíbio.
d) A lua é o satélite natural da terra.
e) A preta é um peixe de água doce.

Questão 5
Ao entrar numa sala, João contou 4 pessoas, incluindo ele.
Todos estavam calçados. Sem contar com ele, quantos sapatos havia na sala?
a) 4.
b) 6.
c) 8.
d) 16.
e) 10.

Questão 6
Complete a frase a seguir: "De Grão em grão..."

a) A galinha fica mais gorda.
b) O grão fica maior.
c) O grãozinho vai ficando gordo.
d) O coqueiro fica cheio.
e) A galinha enche o papo.

Questão 7
O animal já extinto chamado DODO, era:
a) Um réptil.
b) Um dinossauro.
c) Um pássaro.
d) Um peixe que media até 3 metros de comprimento.
e) Uma serpente marinha que se alimentava exclusivamente de algas.

Questão 8
A palavra MARAJÁ quer dizer:
a) Pessoa muito rica.
b) Pessoa que vive sem fazer nada.
c) Pessoa que ganha dinheiro sem trabalhar.
d) Título de nobreza indiano.
e) Espécie de gato selvagem selvagem.

Questão 9
A palavra MARACUTIA tem quantas vogais:
a) 4.
b) 5.
c) 6.
d) 7.
e) 8.

Questão 10
O Carnaval é:
a) 38 dias antes da Páscoa.
b) 39 dias antes da Páscoa.
c) 40 dias antes da Páscoa.
d) 42 dias antes da Páscoa.
e) 45 dias antes da Páscoa.

37

Corrigir prova – outra solução

Matr
≠
0

1) Ler o gabarito do arquivo numa tupla

2) Corrigir prova:

Obter respostas, contando acertos de acordo com o gabarito

```
arq = open("questoes.txt", 'r')
texto = arq.read()
lquestoes = texto.split('#')
tquestoes = tuple(lquestoes)
gab = ('d', 'd', 'a', 'd', 'b', 'e', 'c', 'c', 'c', 'c')
matr = int(input("Sua matrícula? "))
while matr != 0:
    lCorrecao = []
    nota = corrige(gab, tquestoes, lCorrecao)
    exibeCorrecao(matr, nota, lCorrecao)
    matr = int(input("Sua matrícula? "))
```

Será que (e como)
uma função pode
retornar mais que
um valor?????

38

Tuplas como valores de retorno

- O valor de retorno de funções pode ser uma tupla
- Estritamente uma função só pode retornar um valor, mas se o valor for uma tupla, o efeito é o mesmo que retornar vários valores.

Exemplo:

Faça uma função que receba o tempo de duração de um evento em horas e retorne o mesmo valor expresso em números de semanas, dias e horas.

Parâmetros: tempo em horas

Valor de retorno? semanas, dias e horas

39

Tuplas como valores de retorno

```
def periodoEvento(tempohoras):  
    dias = tempohoras//24  
    h = tempohoras%24  
    return (dias//7,dias%7,h)  
  
(semana,dias,horas) = periodoEvento(458)  
print('O evento durou %d semanas, %d dias e %d horas'%(semana,dias,horas))
```

Python 3.5.2 Shell

File Edit Shell Debug Options Window Help

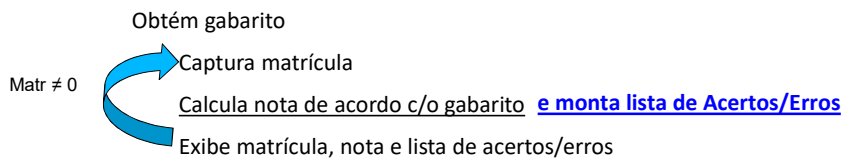
Python 3.5.2 (v3.5.2:4def2a2901a5, Jun 25 2016, 22:01:18) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.

O evento durou 2 semanas, 5 dias e 2 horas

40

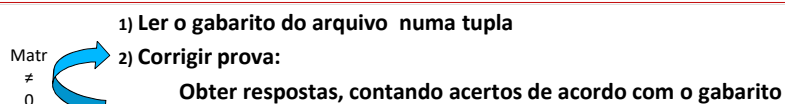
Voltando ao Exercício

A função **corrige** deve retornar duas informações: a nota do aluno e a lista com valores lógicos indicando as questões que acertou e as que errou.



41

Corrigir prova - Principal



```
arq = open("questoes.txt", 'r')
texto = arq.read()
lquestoes = texto.split('#')
tquestoes = tuple(lquestoes)
gab = ('d', 'd', 'a', 'd', 'b', 'e', 'c', 'c', 'c', 'c')
matr = int(input("Sua matrícula? "))
while matr != 0:
    (nota, lCorrecao) = corrige(gab, tquestoes)
    exibeCorrecao(matr, nota, lCorrecao)
    matr = int(input("Sua matrícula? "))
```

42

Revisão: Corrigir prova

```
def corrige(gabarito,tquestoes):  
    nota = 0  
    lista = []  
    for i,el in enumerate(tquestoes):  
        print(el)  
        resp = input("Digite a resposta da questão %d: "%(i+1))  
        if resp == gabarito[i]:  
            nota+=1  
            lista.append(True)  
        else:  
            lista.append(False)  
    return (nota, lista)  
  
def exibeCorrecao(matr,nota,Lista):  
    print("%d - Sua nota é: %d\n"%(matr,nota))  
    print(Lista)
```

43

Revisão: Corrigir prova – Nova versão

Alterar o programa para que a tupla retornada pela função **corrige** contenha a nota e as respostas do aluno.

44

Revisão: Corrigir prova

```
def corrige(gabarito,tquestoes):  
    nota = 0  
    lista = []  
    for i,el in enumerate(tquestoes):  
        print(el)  
        resp = input("Digite a resposta da questão %d: "%(i+1))  
        if resp == gabarito[i]:  
            nota+=1  
        lista.append(resp)  
    return (nota, lista)  
  
def exibeCorrecao(matr,nota,Lista):  
    print("%d - Sua nota é: %d\n"%(matr,nota))  
    print(Lista)
```

45

Exercício

Criar uma tupla com as seguintes perguntas para uma pessoa sobre um crime:

- "Telefonou para a vítima?"
- "Esteve no local do crime?"
- "Mora perto da vítima?"
- "Devia para a vítima?"
- "Já trabalhou com a vítima?"

Seu programa deve exibir estas perguntas a usuários e contabilizar as respostas "Sim" emitindo uma classificação sobre a participação de cada pessoa no crime. Se a pessoa responder positivamente a 2 questões ela deve ser classificada como "Suspeita", entre 3 e 4 como "Cúmplice" e 5 como "Assassina". Caso contrário, ela será classificada como "Inocente".

46

Problema

Faça um programa que pergunte ao usuário dois valores e exiba os números do intervalo formado por eles (inclusive).

Dados necessários: $n1, n2$

Respostas Exibidas? $n1, n1+1, n1+2, \dots, n2$

Como Dados \rightarrow Respostas? Gerar sucessor n e exibir, iniciando em $n1$ até $n2$

47

Desenvolvendo a Solução

Para a solução proposta, $n1 \leq n2$. Caso $n1 > n2$, deve-se trocar os valores das variáveis.

- Com operações de atribuição, é necessário utilizar uma variável temporária:

```
aux = n1
n1  = n2
n2  = aux
```

- Outra forma é utilizar a atribuição de tupla:

$(n1, n2)$	=	$(n2, n1)$
↓		↓
tupla de variáveis		tupla de valores

- ✓ O lado esquerdo é uma tupla de variáveis; o lado direito é uma tupla de valores.
- ✓ Cada valor é atribuído à sua respectiva variável.
- ✓ Todas as expressões do lado direito são avaliadas antes das atribuições.
- ✓ O número de variáveis na esquerda deve ser igual ao de valores na direita.

48

Uma solução

```
n1 = int(input("Digite um número"))
n2 = int(input("Digite outro número"))

if n1>n2:
    (n1,n2)=(n2,n1)

while n1<=n2:
    print("%d "%n1,end='')
    n1+=1
```

49

Exercício

a) Faça a função **divisaoInteira**, que recebe dois valores (o dividendo e o divisor). Esta função retorna:

- uma tupla com o quociente e o resto, se a operação for possível;
- uma tupla vazia quando o resultado é indeterminado ou
- None: quando a divisão é impossível

b) Faça um programa, utilizando a função do item a) para converter um valor lido em BITS, moeda local das ilhas Weblands, na menor quantidade possível de notas de 50, 10, 5 e 1 BITS

50

Exercício

Desenvolva a função **raízes**, que calcula e retorna as raízes **x1** e **x2** de uma equação do segundo grau, do tipo $ax^2 + bx + c = 0$.

a, b e c representam os coeficientes da equação.

Observações:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

- a) Se as raízes forem reais e distintas, a função deve retornar os dois valores e **x1** deve ser associado ao menor valor e **x2** ao maior valor.
- b) Se as raízes forem reais e iguais, a função deve retornar apenas um valor
- c) Se não existirem raízes reais, a função deve retornar None.

Faça um programa completo que pergunte um número indeterminado de coeficientes **a, b e c** ao usuário (o programa deverá terminar quando o valor do coeficiente de **a** for **zero**) e, para cada grupo de coeficientes, chame a função **raízes** e as exiba, caso elas sejam reais, ou a mensagem **Raízes Imaginárias**, caso elas sejam imaginárias.

51

Uma Solução

```
"""Raízes da equação do segundo grau
raízes: recebe os coeficientes de um eq. do seg. grau
ax2 + bx1 + c = 0
"""
def raizes(a,b,c):
    delta = b**2 - 4*a*c
    if delta<0:
        return None
    if delta == 0:
        return (-b/(2*a))
    raiz = math.sqrt(delta)
    x1= (-b + raiz)/(2*a)
    x2= (-b - raiz)/(2*a)
    if (x1 > x2):
        return (x2,x1)
    else:
        return (x1,x2)
```

52

Uma Solução

```
a= float(input("a? "))
while a!=0:
    b= float(input("b? "))
    c= float(input("c? "))
    resp = raizes(1,-10,24)
    if resp:
        if len(resp)==1):
            print("\nRaiz única:%.2f "%resp[0])
        else:
            print("\nx1:%.2f , x2:%.2d"%(resp[0],resp[1]))
    else:
        print("\nRaízes Imaginárias")
    a= float(input("\na? "))
```

53

Problema

Faça uma função que receba uma tupla e retorne a soma de seus elementos

Dados necessários: tupla

Respostas Exibidas? Soma dos elementos

Como Dados → Respostas? Somar todos os elementos

54

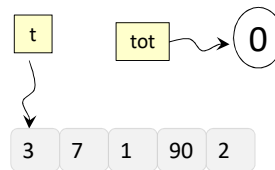
Soma valores de uma tupla

```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el #tot=tot+el  
    return tot  
  
t1 = (3,7,1,90,2)  
t2 = (3,(1,2,98),10)  
print(somaEl(t1))  
print(somaEl(t2))
```

55

Soma valores de t1

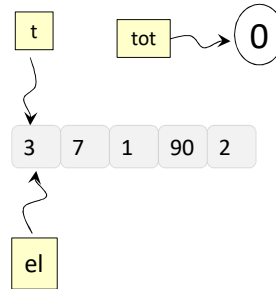
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



56

Soma valores de t1

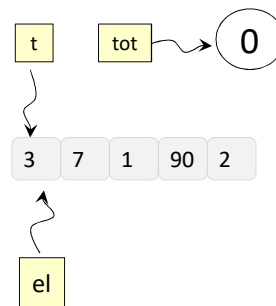
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



57

Soma valores de t1

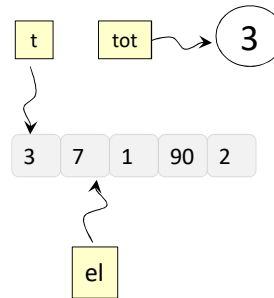
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



58

Soma valores de t1

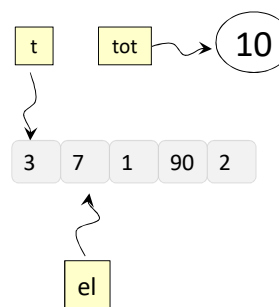
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



59

Soma valores de t1

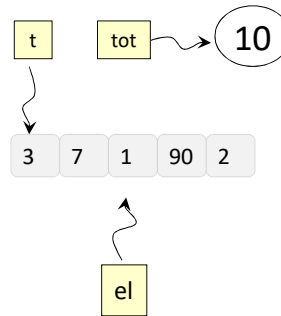
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



60

Soma valores de t1

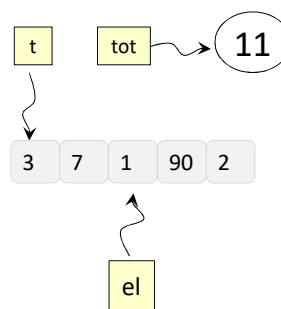
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



61

Soma valores de t1

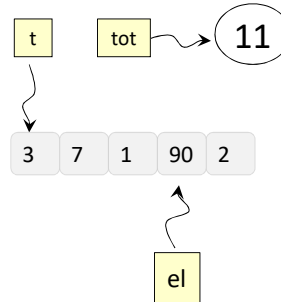
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



62

Soma valores de t1

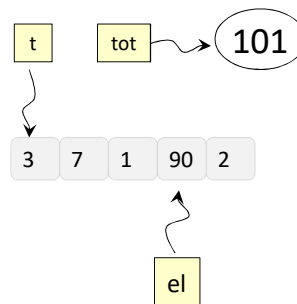
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



63

Soma valores de t1

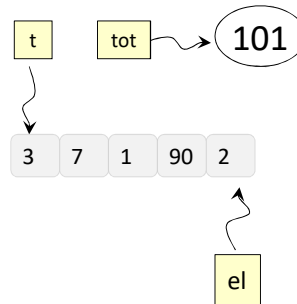
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



64

Soma valores de t1

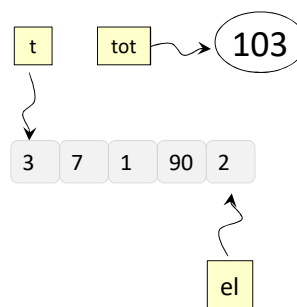
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



65

Soma valores de t1

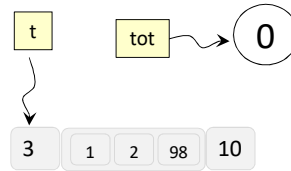
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



66

Soma valores de t2

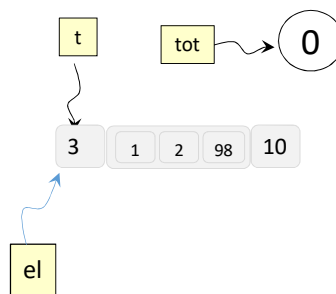
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



67

Soma valores de t2

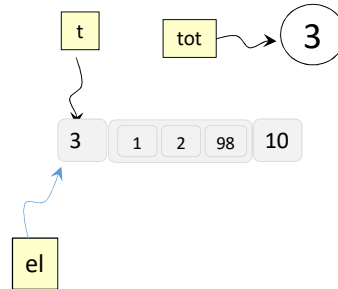
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



68

Soma valores de t2

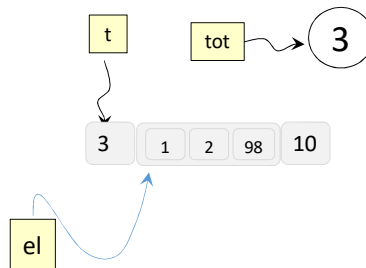
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



69

Soma valores de t2

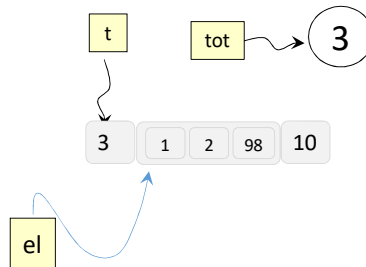
```
def somaEl(t):  
    tot=0  
    for el in t:  
        tot+=el  
    return tot
```



70

Soma valores de t2

```
def somaEl(t):
    tot=0
    for el in t:
        tot+=el
    return tot
```

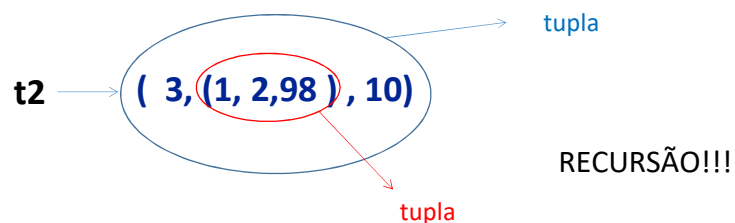


Como o 2º elemento de t2 é uma tupla, não pode ser somado a tot:

TypeError: unsupported operand type(s) for +=: 'int' and 'tuple'

71

Soma valores de t2



Somar Tupla:

Caso 1) o elemento é um inteiro: somar ao total

Caso 2) o elemento é uma tupla: Somar Tupla e somar resultado ao total

72

Soma elementos de uma tupla

```
def somaEl(t):  
    tot=0  
    for el in t:  
        if isinstance(el,int):  
            tot+=el  
        else:  
            tot+=somaEl(el)  
    return tot
```

```
t1=(3,7,1,90,2)  
t2=(3,(1,2,98),10)  
print(somaEl(t1))  
print(somaEl(t2))
```

E se algum elemento
for string ou lista ou
tupla?

73

Soma elementos de uma tupla

```
def somaEl(t):  
    tot=0  
    for el in t:  
        if isinstance(el,int) or isinstance(el,float):  
            tot+=el  
        elif isinstance(el,tuple) or isinstance(el,list):  
            tot+=somaEl(el)  
    return tot
```

```
t1=(3,7,1,90,2)  
t2=(3,(98,2,1),10)  
print(somaEl(t1))  
print(somaEl(t2))
```

Despreza
elemento
se for
string

74

Exercício

Defina uma função recursiva **combina** que receba 2 listas de mesmo tamanho e combina-as em uma lista de tuplas, conforme os exemplos abaixo:

`combina([1,2], [10,20])` retorna `[(1,10),(2,20)]`

`combina([1,2,3], ['a','b','c'])` retorna `[(1,'a'),(2,'b'),(3,'c')]`

75

Uma Solução

```
def combina(l1, l2):  
    if len(l1)==0:  
        return list()  
    lresp = [(l1[0],l2[0])] + combina(l1[1:], l2[1:])  
    return lresp  
  
print(combina([1,2,3], ['a','b','c']))
```

76