

Fundação CECIERJ - Vice Presidência de Educação Superior a Distância
Curso de Tecnologia em Sistemas de Computação
Disciplina: Programação com Interfaces Gráficas
AP2 2º semestre de 2019.

Nome -

Assinatura -

1. (2 pontos) O que é impresso pelo trecho de código abaixo?

```
class C:
    x = 4
    def __init__(self,y):
        self.y = y
    def f(self,a):
        self.y += self.x * a
        self.x -= 1
        return C(self.y)
    def __repr__(self):
        return (str(self.x)+"-"+str(self.y))

print(C("b").f("a-"))
C.x -= 2
print(C([1]).f([2]))
```

2. (2 pontos) O que é impresso pelo trecho de código abaixo?

```
class D:

    def __init__(self,a,b):
        self.d = {}
        for x,y in zip(a,b):
            self.d[x] = self.d.get(x,set())
            self.d[x].add(y)

    def g(self,l):
        return [len (self.d.get(a,set())) for a in l]

print (D("aaaabbccc", "xxxyyyzz").g("abc"))
print (D("abcdabcd",range(10)).g(["a","b",0,1]))
```

Nota: A função zip retorna um iterador de tuplas, onde a i-ésima tupla contém o i-ésimo elemento de cada argumento (que deve ser uma sequência ou objeto iterável). Pode ser usada, por exemplo, para intercalar listas.

3. (6 pontos) Deseja-se simular um jogo de Dominós, e para tanto foi criada a classe Dominos cuja implementação parcial é a seguinte:

```

from random import shuffle
class Dominos:
    "Simula um jogo de dominó"

    def __init__(self):
        "Inicializa um novo jogo"
        ...

    def compra (self):
        """Retira uma peça ainda não sorteada do estoque e a retorna.
        Levanta a exceção ValueError caso não existam mais peças"""
        ...

    def coloca(self,peca,extremidade):
        """Dada uma peça (tupla da forma a,b), e uma extremidade (0 ou 1),
        adiciona a peça ao jogo estendendo a extremidade dada. Retorna
        True se a peça foi colocada com sucesso ou False caso contrário"""
        ...

    def imprime(self):
        """Imprime o jogo até o momento. A impressão tem que ser na forma
        de uma 'escadinha', isto é, as peças colocadas na mesa se
        alternam na orientação horizontal e vertical. O leiaute
        entre uma jogada e a próxima não pode mudar!"""
        ...

```

Nota: lembre que `shuffle(lista)` embaralha os elementos da *lista*! Eis um exemplo de utilização:

```

d = Dominos()
for i in range(6):
    p = d.compra()
    print ("Compra:", p)
    if d.coloca(p,0) or d.coloca(p,1): d.imprime()

```

que imprime:

```

Compra: (5, 6)
56
Compra: (1, 5)
1
5
56
Compra: (2, 2)
Compra: (3, 4)
Compra: (0, 4)
Compra: (1, 3)
31
1
5
56

```