

Travessia do Rio: Lobo-Cabra-Alfafa

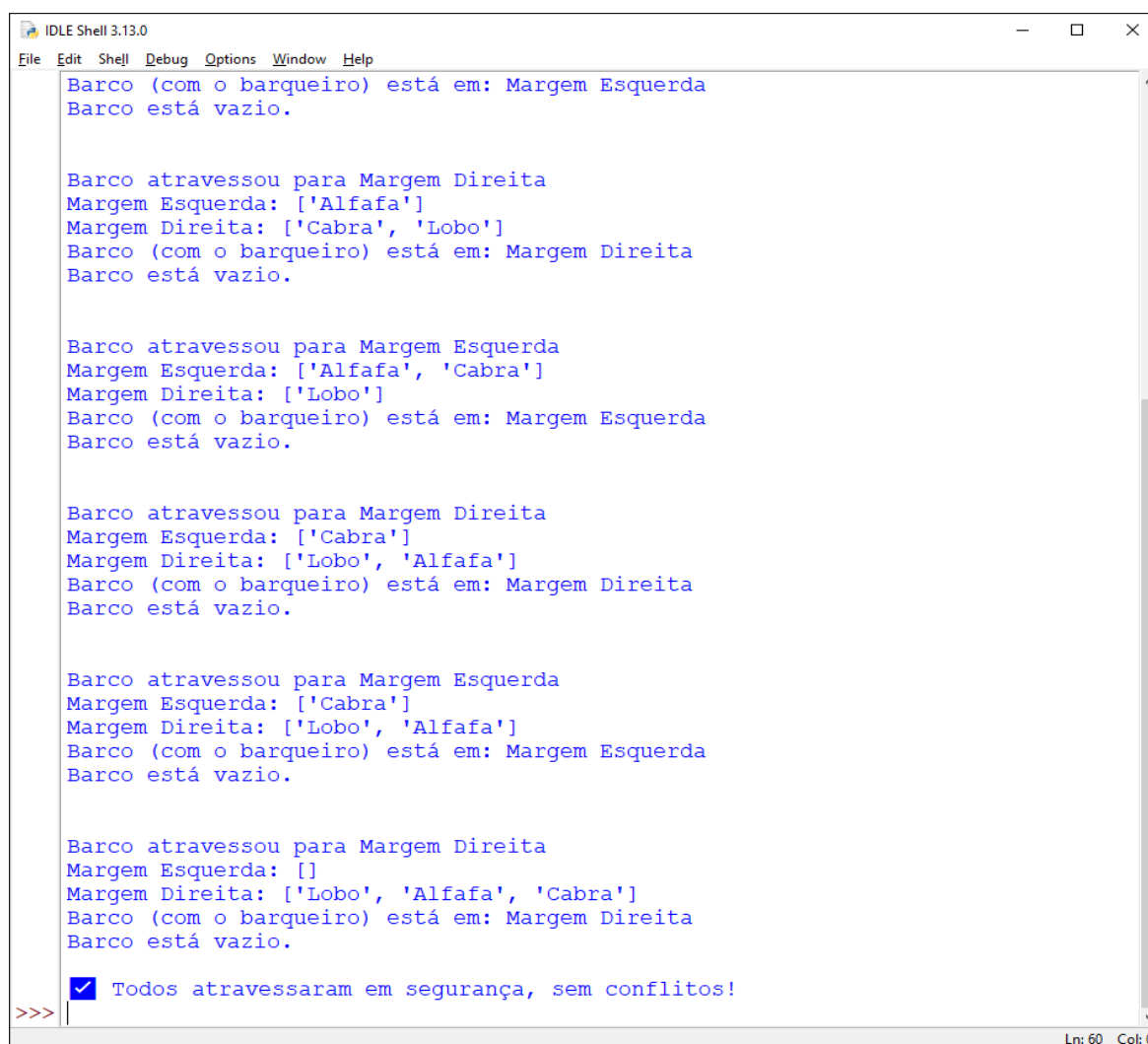
Mário Leite

Este é um problema de desafio lógico muito difundido, que exige certa habilidade para resolver. Considere o seguinte cenário...

Estão na margem de um rio quatro personagens: um lobo, um monte de alfafa, um barqueiro e uma cabra; um ao lado do outro e nesta ordem. O desafio lógico é fazer com que o barqueiro atravesse os três outros personagens para a outra margem, em segurança, de modo que não ocorra nenhum conflito entre eles: por exemplo, o lobo não pode ficar sozinho com a cabra e nem a cabra pode ficar sozinha com a alfafa. As regras para este desafio são as seguintes:

- * Com o barqueiro na mesma margem não há nenhum conflito entre os personagens.
- * O barqueiro pode levar apenas um dos três personagens por vez no barco.
- * Se o lobo for deixado sozinho com a cabra, ele come a cabra.
- * Se a caba for deixada sozinha com a alfafa, ela come a alfafa.

O programa **“TravessiaPerigosa”** mostra a descrição de uma solução do problema passo a passo, prevenindo os possíveis conflitos que porventura possam existir. A **figura 1** mostra saída do programa.



```
IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help

Barco (com o barqueiro) está em: Margem Esquerda
Barco está vazio.

Barco atravessou para Margem Direita
Margem Esquerda: ['Alfafa']
Margem Direita: ['Cabra', 'Lobo']
Barco (com o barqueiro) está em: Margem Direita
Barco está vazio.

Barco atravessou para Margem Esquerda
Margem Esquerda: ['Alfafa', 'Cabra']
Margem Direita: ['Lobo']
Barco (com o barqueiro) está em: Margem Esquerda
Barco está vazio.

Barco atravessou para Margem Direita
Margem Esquerda: ['Cabra']
Margem Direita: ['Lobo', 'Alfafa']
Barco (com o barqueiro) está em: Margem Direita
Barco está vazio.

Barco atravessou para Margem Esquerda
Margem Esquerda: ['Cabra']
Margem Direita: ['Lobo', 'Alfafa']
Barco (com o barqueiro) está em: Margem Esquerda
Barco está vazio.

Barco atravessou para Margem Direita
Margem Esquerda: []
Margem Direita: ['Lobo', 'Alfafa', 'Cabra']
Barco (com o barqueiro) está em: Margem Direita
Barco está vazio.

☒ Todos atravessaram em segurança, sem conflitos!
>>>
```

Figura 1 - Uma solução do problema

```
'''
```

TravessiaPerigosa.py

Resolve o clássico problema-desafio em que um barqueiro, um lobo, uma cabra e um fardo de alfafa, com o desafio atravessar todos de uma margem do rio para a outra sem que haja algum tipo de conflito.

```
'''
```

```
class ClsPersonagem:
```

```
    def __init__(self, nome):
        self.nome = nome
```

```
    def __repr__(self):
        return self.nome
```

```
#-----
```

```
class ClsMargem:
```

```
    def __init__(self, nome):
        self.nome = nome
        self.itens = []
```

```
    def Adicionar(self, item):
        self.itens.append(item)
```

```
    def Remover(self, item):
        self.itens.remove(item)
```

```
    def __repr__(self):
        return f"{self.nome}: {[item.nome for item in self.itens]}"
```

```
#-----
```

```
class ClsBarco:
```

```
    def __init__(self, margem_inicial):
        self.local = margem_inicial
        self.passageiro = None
```

```
    def Embarcar(self, item):
        if(self.passageiro is None):
            self.local.Remover(item)
            self.passageiro = item
        else:
            print("O barco já tem um passageiro!")
```

```
    def Desembarcar(self):
        if(self.passageiro):
            self.local.Adicionar(self.passageiro)
            self.passageiro = None
```

```
    def Atravessar(self, nova_margem):
        self.local = nova_margem
        print(f"\nBarco atravessou para {nova_margem.nome}")
```

```
#-----
```

```

class ClsTravessia:
    def __init__(self):
        self.lobo = ClsPersonagem("Lobo")
        self.cabra = ClsPersonagem("Cabra")
        self.alfafa = ClsPersonagem("Alfafa")
        self.margem_esquerda = ClsMargem("Margem Esquerda")
        self.margem_direita = ClsMargem("Margem Direita")
        self.margem_esquerda.Adicionar(self.lobo)
        self.margem_esquerda.Adicionar(self.cabra)
        self.margem_esquerda.Adicionar(self.alfafa)
        self.barco = ClsBarco(self.margem_esquerda)

    def ChecarEstadoPerigoso(self):
        for margem in [self.margem_esquerda, self.margem_direita]:
            LstNomes = [item.nome for item in margem.itens]
            if (self.barco.local != margem): #verifica se o barqueiro NÃO está presente
                if ("Lobo" in LstNomes and "Cabra" in LstNomes):
                    return f"⚠ Conflito! O lobo comeu a cabra na {margem.nome}!"
                elif ("Cabra" in LstNomes and "Alfafa" in LstNomes):
                    return f"⚠ Conflito! A cabra comeu a alfafa na {margem.nome}!"
        return None

    def MostrarEstado(self):
        print(self.margem_esquerda)
        print(self.margem_direita)
        print(f"Barco (com o barqueiro) está em: {self.barco.local.nome}")
        if (self.barco.passageiro):
            print(f"Passageiro no barco: {self.barco.passageiro.nome}")
        else:
            print("Barco está vazio.")
        print()
        conflito = self.ChecarEstadoPerigoso()
        if (conflito):
            print(conflito)

    def Jogar(self):
        self.MostrarEstado()

        # 1: Leva a Cabra
        self.barco.Embarcar(self.cabra)
        self.barco.Atravessar(self.margem_direita)
        self.barco.Desembarcar()
        self.MostrarEstado()

        # 2: Volta sozinho
        self.barco.atravessar(self.margem_esquerda)
        self.MostrarEstado()

        # 3: Leva o Lobo
        self.barco.Embarcar(self.lobo)
        self.barco.Atravessar(self.margem_direita)
        self.barco.Desembarcar()
        self.MostrarEstado()

        # 4: Traz a Cabra de volta
        self.barco.Embarcar(self.cabra)
        self.barco.Atravessar(self.margem_esquerda)
        self.barco.Desembarcar()
        self.MostrarEstado()

```

```

# 5: Leva a Alfafa
self.barco.Embarcar(self.alfafa)
self.barco.Atravessar(self.margem_direita)
self.barco.Desembarcar()
self.MostrarEstado()

# 6: Volta sozinho
self.barco.Atravessar(self.margem_esquerda)
self.MostrarEstado()

# 7: Leva a Cabra novamente
self.barco.Embarcar(self.cabra)
self.barco.Atravessar(self.margem_direita)
self.barco.Desembarcar()
self.MostrarEstado()

print("✅ Todos atravessaram em segurança, sem conflitos!")
#=====
# Executar o jogo
jogo = ClsTravessia()
jogo.Jogar()
#Fim do programa "TravessiaPerigosa" -----

```