

## O Problema das Lâmpadas

Mário Leite

...

Considere a seguinte situação: *simular uma casa inteligente em que é possível acender/apagar uma lâmpada em um dos nove quartos da casa, mantendo o controle dos status das lâmpadas de cada quarto. Considere os nove quartos compondo uma matriz 3x3, em que uma lâmpada acesa tenha o valor 1 e apagada o valor 0. Depois de montar a matriz verificar o status da lâmpada de determinado quarto; se estiver acesa deverá ser apagada e se estiver apagada deverá ser acesa. Fazer o diagrama de blocos e em seguida codificar o programa em Python.*

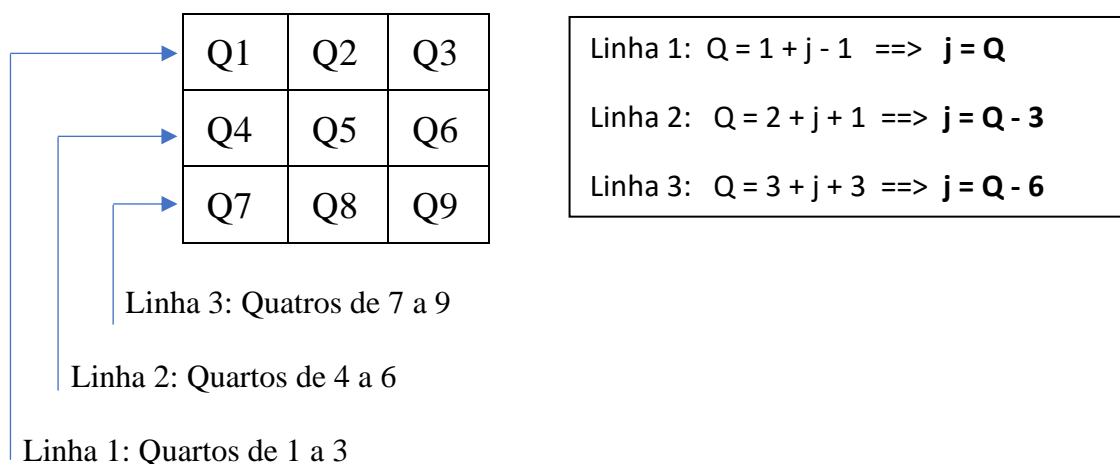
Este poderia ser um problema a ser resolvido num trabalho pedido num curso de Programação, valendo nota para compor a média.

Resumindo: uma solução poderá ser a seguinte...

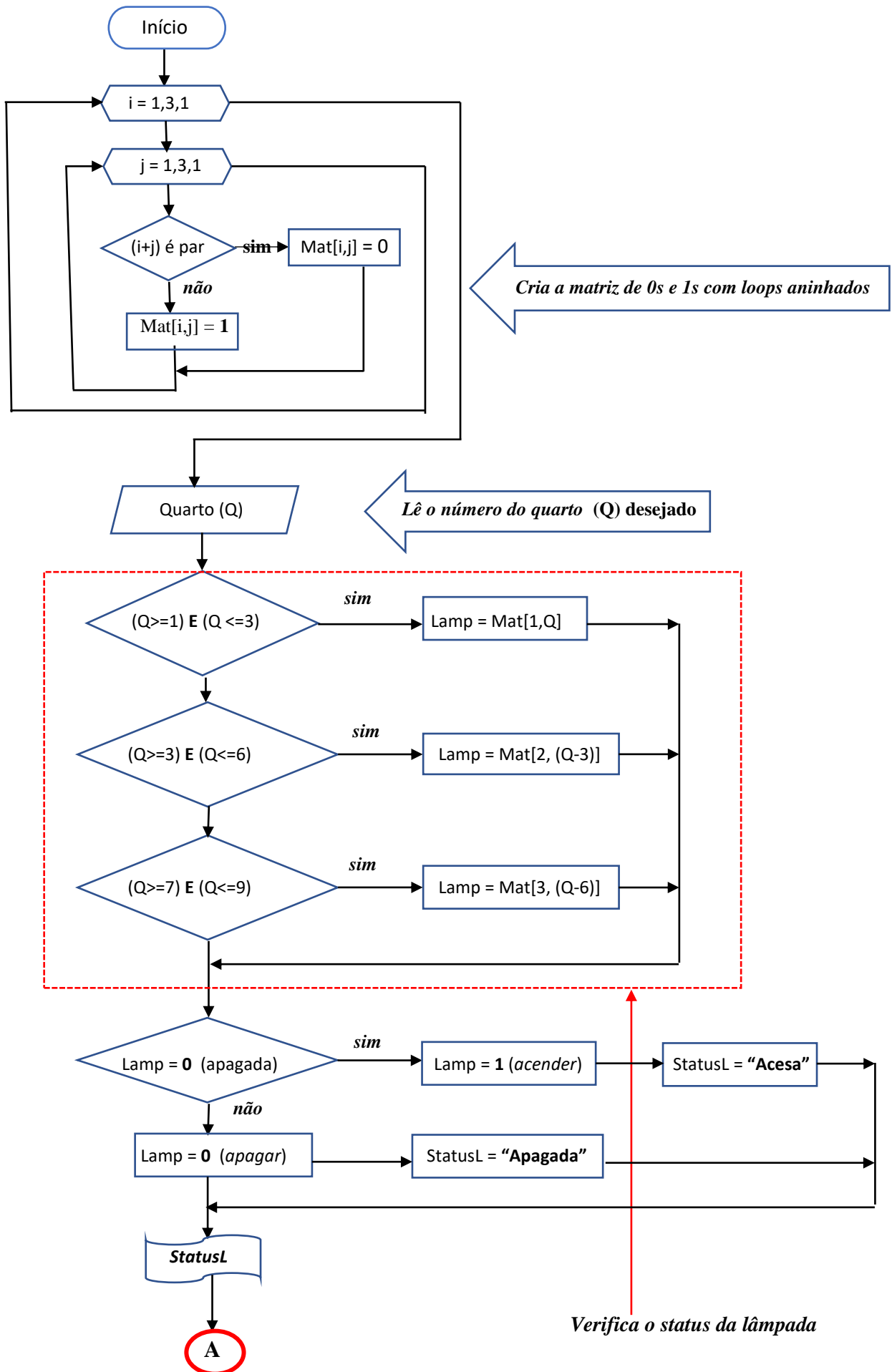
- 1) Focar nos nove quartos da casa.
- 2) Considerar uma matriz 3x3 em que cada elemento seja um quarto com uma única lâmpada.
- 3) Os *status* das lâmpadas devem ser iniciadas com 0's e 1's com alguma lógica, e não aleatoriamente.
- 4) Se a lâmpada estiver acesa seu *status* é 1; se estiver apagada seu *status* é 0.
- 5) Depois de iniciada a matriz das lâmpadas deve ser feita a verificação/alteração do *status* de uma delas num quarto. Após a alteração mostrar a nova configuração das nove lâmpadas.

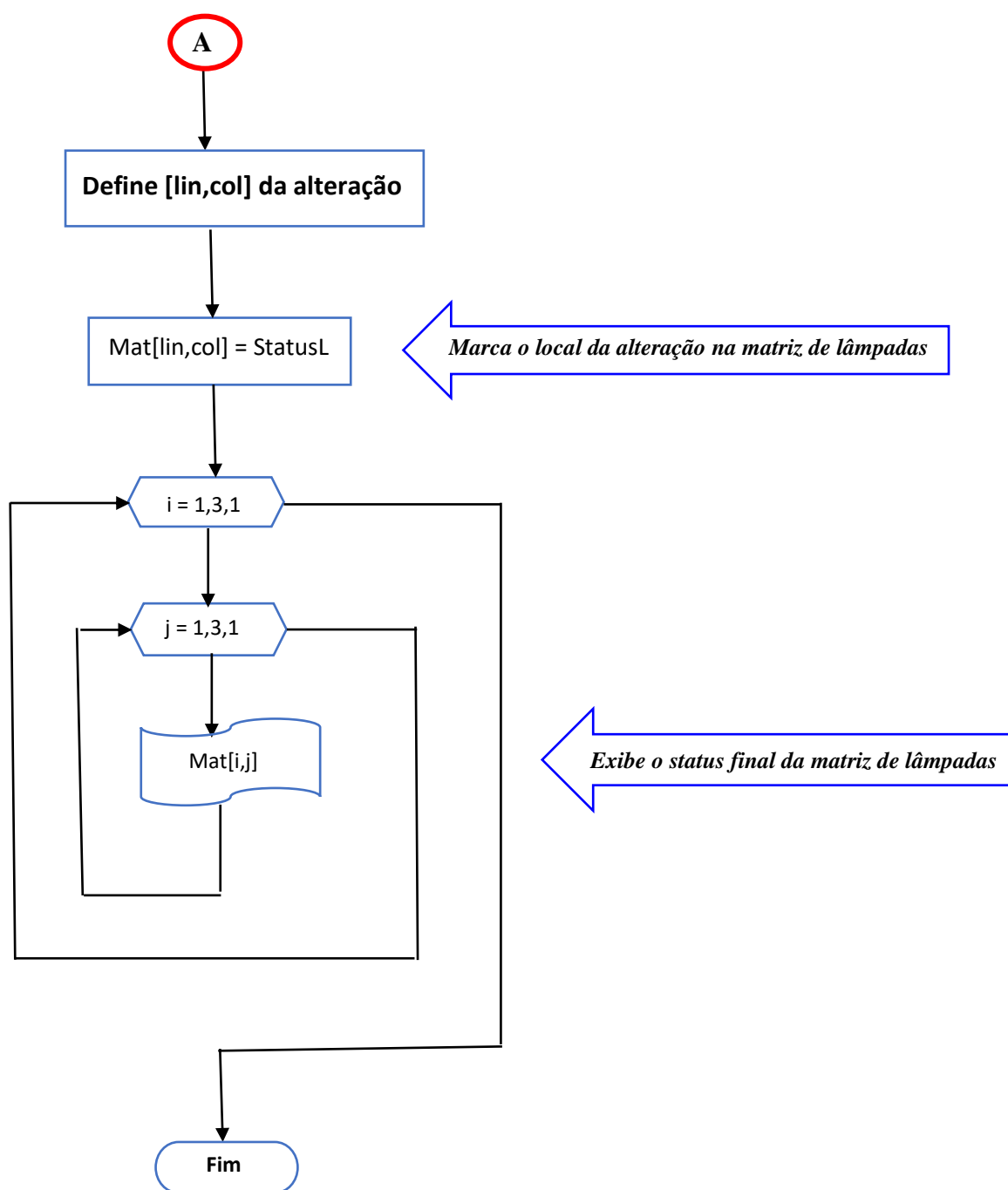
### Lógica aplicada para verificar o *status* da lâmpada de um quarto

Lendo o número do quarto **Q** sabe-se a **linha (i)** da matriz de lâmpadas. Em função destes dois dados é possível descobrir a **coluna (j)** o que permite saber o valor do elemento **[i, j]** da matriz que representa o *status* da lâmpada de qualquer quarto.



**Nota:** Embora o uso de “Diagramas de Blocos” esteja um pouco “fora de moda” (o Visualg é bem melhor para criar a solução), este problema foi formulado deste jeito por um aluno de graduação em “Engenharia Civil” da UFRR. Na codificação em Python os índices **[i, j]** da matriz tiveram que ser decrementados de 1 para atender as exigências da linguagem, cujos valores começam por 0.





**Figura 1 - Diagrama de blocos do programa**

```
ControleLampadas x
"D:\Cantinho da Programação\Códigos\Python\ControleLampadas\venv\Scripts\python.exe"

Situação inicial da matriz de lâmpadas dos quartos:
0  1  0
1  0  1
0  1  0

Entre com número do quarto a ser verificado: 5
Lâmpada do quarto 5 estava apagada e foi acesa.

Situação final da matriz de lâmpadas dos quartos:
0  1  0
1  1  1
0  1  0

Process finished with exit code 0
|
```

**Figura 2 - Simulação para a lâmpada do quarto 5**

### Código do programa

```
'''
Programa "ControleLampadas.py"
Faz o controle das lâmpadas de nove quartos, definidos numa matriz 3x3.
Se a lâmpada estiver acesa deve ser apagada e vice versa.
Em Python
Autor: Mário Leite
-----
'''

# Início -----
print("")
i = 0
j = 0
# Define a matriz de status das lâmpadas dos quartos em 0's e 1's
Mat = [] #inicia a lista
for i in range(3):
    #Cria a linha i
    linha = [] #reestabelece lista vazia
    for j in range(3):
        lix = (i + j) % 2
        if (lix == 0):
            linha.append('0')
        else:
            linha.append('1')
    #Adiciona linha à matriz
    Mat.append(linha)
print("")

# Exibe a matriz com os status iniciais das lâmpadas em cada quarto
print("Situação inicial da matriz de lâmpadas dos quartos:")
for i in range(3):
    for j in range(3):
        print(Mat[i][j], " ", end="")
    print("")
print("")
```

```

# Entrada e validação do quarto desejado
Q = 0
while((Q<1) or (Q>9)):
    Q = int(input("Entre com número do quarto a ser verificado: "))
# Verifica os status das lâmpadas de cada quarto
if((Q>=1) and (Q<=3)): #primeira linha
    lamp = Mat[0][(Q-1)]
    lin = 0
    col = (Q-1)
if((Q>=4) and (Q<=6)): #segunda linha
    lamp = Mat[1][(Q-4)]
    lin = 1
    col = (Q-4)
if((Q>=7) and (Q<=9)): #terceira linha
    lamp = Mat[2][(Q-7)]
    lin = 2
    col = (Q-7)

# Executa a ação desejada (acender/apagar) na matriz
if ((lin + col) % 2 == 0): #estava apagada
    Mat[lin][col] = '1' # ==> ACENDER
else: #estava acesa
    Mat[lin][col] = '0' # ==> APAGAR

# Exibe o status atual da lâmpada do quarto e o modifica
if(lamp=='0'):
    lamp = '1'
    print(f'Lâmpada do quarto {Q} estava apagada e foi acesa.')
else:
    lamp = '0'
    print(f'Lâmpada do quarto {Q} estava acesa e foi apagada.')

print("")

# Mostra a situação final das lâmpadas dos quartos
print("Situação final da matriz de lâmpadas dos quartos:")
for i in range(3):
    for j in range(3):
        print(Mat[i][j], " ", end="")
    print("")
#FimPrograma-----

```