

**Algoritmo e Programação I – 2023/2**  
**AULA 13**

**1 - Lista de listas – lista como matriz bidimensional**

■ **Uma lista pode armazenar valores de tipos diferentes.**

A lista a seguir armazena os dados de um aluno, que consiste em CGU, nome e nota:

```
aluno = [330141, 'Ana Maria', 5.8]
print(aluno)
```

⇒

aluno[0] □ 330141 (inteiro)  
aluno[1] □ 'Ana Maria' (alfanumérico/String)  
aluno[2] □ 5.8 (real)



Exemplo de execução do código:

[330141, 'Ana Maria', 5.8]

Como seria a entrada de dados a partir de valores digitados pelo usuário?

```
aluno = []
aluno.append(int(input('CGU: ')))
aluno.append(input('Nome: '))
aluno.append(float(input('Média final: ')))

print(aluno)
```



Exemplo de execução do código:

CGU: 221133

Nome: Ana Banana

Média final: 9.5

[221133, 'Ana Banana', 9.5]

E se fosse necessário armazenar os dados de mais de um estudante?

■ **Listas aninhadas** □ Uma lista pode armazenar outras listas, ou seja, é possível inserir listas dentro de listas.

No exemplo a seguir, o usuário entrará com os dados de 3 alunos e, depois disso, imprimirá os dados de cada aluno em uma linha:

```
listaAlunos = [] #guardará os dados de todos os alunos
i = 0
while i < 3:
    novoAluno = [] #lista que guardará os dados de um aluno (serão criadas 3)
    novoAluno.append(int(input('CGU: ')))
    novoAluno.append(input('Nome: '))
    novoAluno.append(float(input('Média final: ')))
    print('-----')
    listaAlunos.append(novoAluno) #cada lista novoAluno preenchida é armazenada em
listaAlunos
    i = i + 1

#imprimir lista
print("\n-----\n")
print('Lista de alunos: ')
```

```

i = 0
while i < 3:
    print("CGU:", listaAlunos[i][0], "Nome:", listaAlunos[i][1], "Média:", listaAlunos[i][2])
    i = i + 1

```



Exemplo de execução do código:

```

CGU: 112233
Nome: Mickey Mouse
Média final: 9.5
-----
CGU: 224433
Nome: Minnie Mouse
Média final: 9.8
-----
CGU: 338844
Nome: Pateta
Média final: 10.0
-----
Lista de alunos:
CGU: 112233 Nome: Mickey Mouse Média: 9.5
CGU: 224433 Nome: Minnie Mouse Média: 9.8
CGU: 338844 Nome: Pateta Média: 10.0

```

Outro exemplo de utilização de lista dentro de listas: o programa a seguir lê os dados de frutas vendidas em uma banca e armazena em uma lista, sendo que é perguntado se deseja continuar ao final de cada inserção. Para cada fruta foi solicitado o nome, o valor do quilo e a quantidade em estoque (em kg).

Depois da entrada de dados, foi criada uma estrutura de repetição **while**, que percorre a lista e, em cada iteração, apresenta os dados da fruta e valor que receberá caso todas as frutas sejam vendidas (quantidade em estoque \* preço da fruta).

```
listaFrutas=[]
```

```
continuar = 's'
```

```

while continuar == 's':
    novaLinha = []
    novaLinha.append(input('Nome da fruta:'))
    novaLinha.append(float(input('Preço do kg:')))
    novaLinha.append(float(input('Quantidade em estoque:')))
    listaFrutas.append(novaLinha)
    continuar = input('Deseja continuar a entrada de dados? ')

```

```
#imprimir lista
```

```
print('\n\nLista de frutas:')
```

```
i = 0
```

```
while i < len(listaFrutas):
```

```
    valor = listaFrutas[i][1]*listaFrutas[i][2]
```

```
    print(listaFrutas[i][0], 'Preço:', listaFrutas[i][1], 'Qt. estoque:', listaFrutas[i][2], 'kg. Valor que receberá: ', valor)
```

```
    i = i + 1
```



Exemplo de execução do código:

Nome da fruta:banana

Preço do kg:2.3

Quantidade em estoque:20

Deseja continuar a entrada de dados? s

Nome da fruta:goiaba

Preço do kg:5.5

Quantidade em estoque:10

Deseja continuar a entrada de dados? n

Lista de frutas:

banana Preço: 2.3 Qt. estoque: 20.0 kg. Valor que receberá: 46.0

goiaba Preço: 5.5 Qt. estoque: 10.0 kg. Valor que receberá: 55.0