Matrizes

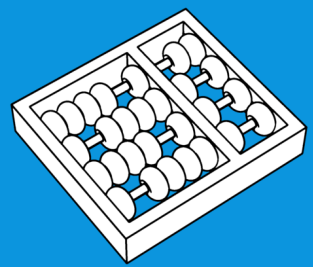
MC102 - Algoritmos e Programação de Computadores

04/24

15

Santiago Valdés Ravelo
https://ic.unicamp.br/~santiago/
ravelo@unicamp.br

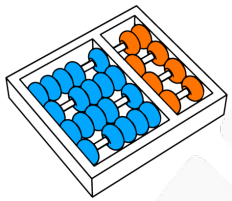
UNICAMP



"Matrizes agem. Elas não apenas ficam lá."

Gilbert Strang.

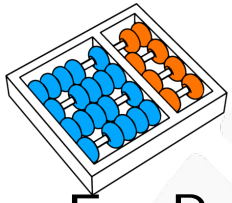
Dúvidas da aula anterior



Dúvidas selecionadas

- Não entendi muito bem qual é a diferença entre os métodos add() e update() ao adicionar elementos a um conjunto. E em que situações cada um deles seria mais apropriado?
- Foi falado em aula que ao percorrer um conjunto, a ordem é aleatória. Como é implementada a "aleatoriedade"no computador ?
- Professor, ainda não ficou muito claro pra mim a diferença, na aplicabilidade, entre dicionários, conjuntos, listas.
- Se os conjuntos não têm índices, o que poderíamos fazer para pegar o primeiro elemento do conjunto, como naquele caso em que queremos encontrar o menor elemento?
- Pelo que foi falado até agora Python tem quatro tipos de coleções de dados: listas, dicionários, conjuntos e tuplas. Será que todas as linguagens tem todos esses tipos ou essa variedade é exclusiva de Python?
- Porque não é possível adicionar tuplas em um conjunto?
- Consigo saber o tamanho do meu set?
- Santiago, o que faz de cada classe ser "hashable"ou não?
- Nos casos em que usamos os métodos de diferença (s1 -= s2), subconjunto (s1 <= s2) e superconjunto (s1 >= s2), é a mesma coisa que a utilização de operadores aritméticos e os de comparação?
- Em outras linguagens, a estrutura set é sempre ordenada. Existe alguma forma de fazer com que set seja ordenado em python?
- Em conjuntos, qual a diferença entre set1 < set2 e set1 <= set2?
- É possível fazer com que conjuntos aceitem apenas um tipo de dado, como apenas int ou apenas string?
- Internamente, os sets são árvores binárias ou usam hashes?

Matrizes



Matrizes

Em Python, matrizes são representadas como listas de listas.

Isto é, podemos representar a matriz:

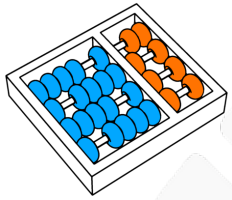
$$\begin{bmatrix} 7 & 0 & 2 & 3 \\ 3 & 1 & 4 & 2 \\ 0 & 3 & 2 & 7 \end{bmatrix}$$

Como **m = [[7, 0, 2, 3], [3, 1, 4, 2], [0, 3, 2, 7]]**.

- Isto é uma lista de linhas da matriz.
- Onde cada linha é, também, uma lista.

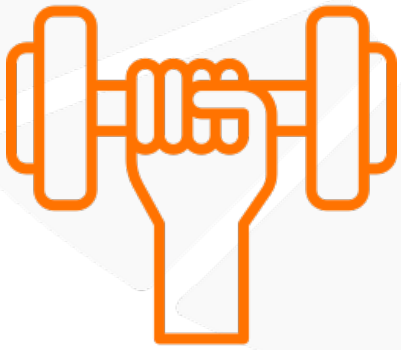
A célula da linha **i** coluna **j** é acessada escrevendo **m[i][j]**.

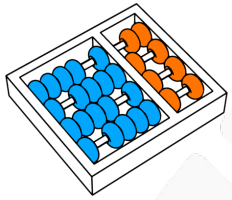
- para leitura ou escrita
- Ex: **m[0][0]** é **7**, **m[0][1]** é **0**, **m[1][0]** é **3**, etc



Matrizes

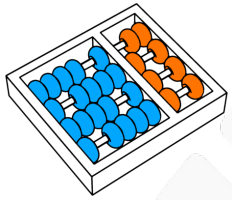
Vamos fazer alguns exercícios?





Exercícios

1. Faça uma função que dados inteiros **n** e **m**, devolve uma matriz **n**×**m** (isto é, **n** linhas e **m** colunas) com algum valor inicial dado (zero por padrão).
2. Faça uma função que, dado um inteiro **n**, devolve a matriz identidade **n**×**n**.
3. Faça uma função que, dada uma matriz, imprime a matriz (em sua representação usual).
4. Faça uma função que, dada uma matriz M e um escalar λ, calcula $\lambda \cdot M$.
5. Faça uma função que soma duas matrizes.
6. Faça uma função que transpõem uma matriz.



Multiplicação de Matrizes

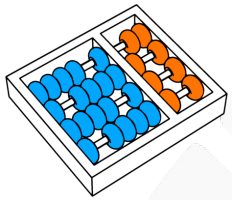
Sejam A uma matriz $n \times m$ e B uma matriz $m \times p$.

O produto $C = A \cdot B$ é a matriz $n \times p$ tal que:

$$\mathbf{C_{i,j}} = \sum_{\mathbf{k=1}}^{\mathbf{m}} \mathbf{A_{i,k}} \cdot \mathbf{B_{k,j}}.$$

Exercício: Faça uma função que multiplica duas matrizes.

Um aplicação



Aplicação: Imagens

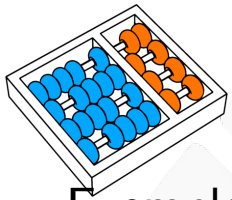
Imagens são matrizes de pixels

- Preto-e-Branco: basta um bit por pixel.
- Escala de cinza: um valor entre 0 e 255.
- RGB (24-bit): cada posição tem três valores entre 0 e 255.
- Entre outros modelos.

Existem vários formatos de arquivo de imagem:

- jpg, png, gif, tiff, etc...

Vamos usar um particularmente fácil de trabalhar...



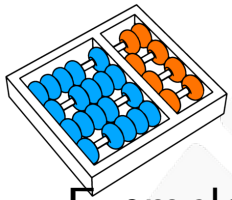
Um arquivo pbm (preto-e-branco)

Exemplo do arquivo:

```
P1
25 7
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 1 0 1 1 1 1 1 0 1 0 1 1 1 1 0 1 1 1 1 0
0 1 1 0 1 1 0 1 0 0 0 0 1 0 1 0 0 1 0 0 0 0 1 0 0
0 1 0 1 0 1 0 1 0 0 0 0 1 0 1 0 0 1 0 1 1 1 1 0 0
0 1 0 0 0 1 0 1 0 0 0 0 1 0 1 0 0 1 0 1 0 0 0 0 0
0 1 0 0 0 1 0 1 1 1 1 1 0 1 0 1 1 1 1 0 1 1 1 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Formato:

- Sempre começa com **P1**.
- Na segunda linha, temos o número de colunas e de linhas.
- E colocamos a matriz de bits separados por espaço:
  - bit 1 indica pixel preto.



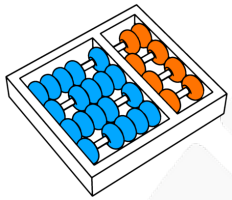
Um arquivo pbm (preto-e-branco)

Exemplo do arquivo:

```
P1
25 7
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 1 0 1 1 1 1 0 1 0 1 1 1 1 0 1 1 1 1 0
0 1 1 0 1 1 0 1 0 0 0 0 1 0 1 0 0 1 0 0 0 0 1 0
0 1 0 1 0 1 0 1 0 0 0 0 1 0 1 0 0 1 0 1 1 1 1 0
0 1 0 0 0 1 0 1 0 0 0 0 1 0 1 0 0 1 0 1 0 0 0 0
0 1 0 0 0 1 0 1 1 1 1 0 1 0 1 1 1 1 0 1 1 1 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

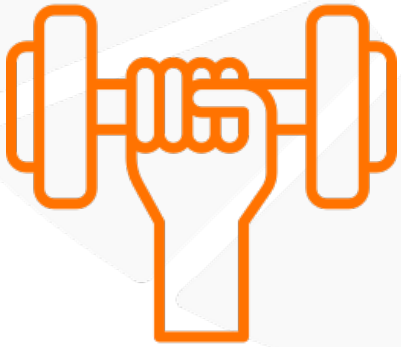
Resultado:

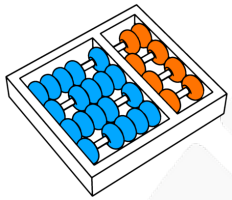
MC102



Imagens

Vamos fazer alguns exercícios?

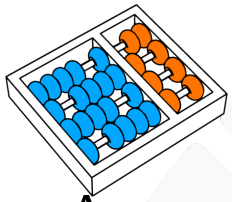




Exercícios

1. Faça uma função que lê do terminal o conteúdo de um arquivo **pbm**.
2. Faça uma função que, dada uma matriz, escreve (no terminal) o conteúdo de um arquivo **pbm**.
3. Faça uma função que, dada uma matriz de **0**'s e **1**'s, nega a matriz, isto é, posições que eram **0** viram **1** e vice-versa.
4. Combine os três exercícios anteriores para inverter as cores de uma imagem **pbm**.

Múltiplas dimensões



Matrizes d-dimensionais

As matrizes que vimos têm duas dimensões:

- linhas e colunas.

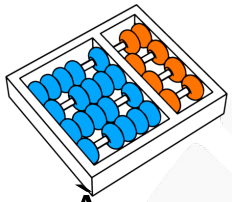
Mas podemos querer ter matrizes com mais dimensões. . .

- Ex: Um vídeo é uma sequência de imagens.
- Ex: Em imagens coloridas, em cada linha/coluna, temos três valores.

Mas isso é fácil de resolver!

- Quando tínhamos uma dimensão usamos listas.
- Quando tínhamos duas dimensões usamos listas de listas.
- Quando temos três dimensões usamos listas de listas de listas!
- E assim por diante!

Linearização



Linearizando índices

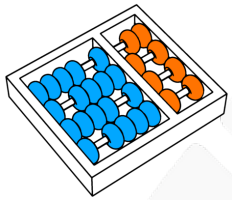
As vezes precisamos usar uma lista ao invés de listas de listas:

- Por questões de eficiência.

Para tanto, precisamos de uma função bijetora entre as posições (i, j) da matriz e as posições k da lista:

- Dada uma posição da matriz, queremos a posição da lista.
  - Ex: na hora de acessar o valor.
- E dada uma posição da lista, queremos a posição da matriz.
  - Ex: na hora de imprimir a matriz.

$$m = \begin{bmatrix} 7 & 0 & 2 \\ 3 & 1 & 4 \\ 0 & 3 & 2 \end{bmatrix} \quad \longleftrightarrow \quad l = \begin{bmatrix} 7 & 0 & 2 & 3 & 1 & 4 & 0 & 3 & 2 \end{bmatrix}$$



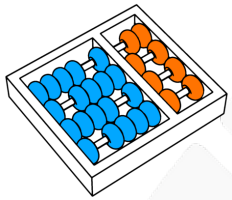
Linearizando índices

$$m = \begin{bmatrix} 7 & 0 & 2 \\ 3 & 1 & 4 \\ 0 & 3 & 2 \end{bmatrix} \quad \longleftrightarrow \quad l = \begin{bmatrix} 7 & 0 & 2 & 3 & 1 & 4 & 0 & 3 & 2 \end{bmatrix}$$

A posição:

- **(0,0)** deve ir para o índice 0.
- **(0,1)** deve ir para o índice 1.
- **(0, m - 1)** deve ir para o índice $m - 1$.
- **(1,0)** deve ir para o índice m.
- **(1,1)** deve ir para o índice $m + 1$.
- **(1, m - 1)** deve ir para o índice $2m - 1$.
- **(i, j)** deve ir para o índice $i \cdot m + j$.

Mas como voltar da lista para a matriz?



Linearizando índices

$$m = \begin{bmatrix} 7 & 0 & 2 \\ 3 & 1 & 4 \\ 0 & 3 & 2 \end{bmatrix} \quad \longleftrightarrow \quad l = \begin{bmatrix} 7 & 0 & 2 & 3 & 1 & 4 & 0 & 3 & 2 \end{bmatrix}$$

A posição (i, j) deve ir para o índice $i \cdot m + j$.

Mas como ir da posição k da lista para a matriz?

- Precisamos saber quantos linhas completas formamos:
  - Isto é, $i = k \,//\, m$.
- E quantas colunas sobraram:
  - Isto é, $j = k \,\%\, m$.

Exercício: Faça um programa que lê duas matrizes, soma as duas e imprime o resultado usando linearização de índices.

Desafio: Linearize uma matriz tridimensional.

Matrizes

MC102 - Algoritmos e Programação de Computadores

04/24

15

Santiago Valdés Ravelo
https://ic.unicamp.br/~santiago/
ravelo@unicamp.br

UNICAMP