

## Manipulação de arquivos

Neste arquivo será descrito formas de manipulação de arquivos em *Python*, ou seja, como realizar leitura de arquivos, transformando o conteúdo do arquivo de texto em dados do programa e também a escrita de arquivos, convertendo dados do programa para um arquivo de texto. Especificamente o contexto tratado aqui será de matrizes.

### 1 Leitura de Arquivo

Suponha que existe um arquivo com nome **matriz1.txt** onde cada linha deste arquivo contém os números que correspondem à linha da matriz. Por exemplo, a matriz  $A$  abaixo representada no arquivo texto seria:

$$A = \begin{pmatrix} 35 & 1 & 6 & 26 & 19 & 24 \\ 3 & 32 & 7 & 21 & 23 & 25 \\ 31 & 9 & 2 & 22 & 27 & 20 \\ 8 & 28 & 33 & 17 & 10 & 15 \\ 30 & 5 & 34 & 12 & 14 & 16 \\ 4 & 36 & 29 & 13 & 18 & 11 \end{pmatrix} \quad \begin{matrix} 35 & 1 & 6 & 26 & 19 & 24 \\ 3 & 32 & 7 & 21 & 23 & 25 \\ 31 & 9 & 2 & 22 & 27 & 20 \\ 8 & 28 & 33 & 17 & 10 & 15 \\ 30 & 5 & 34 & 12 & 14 & 16 \\ 4 & 36 & 29 & 13 & 18 & 11 \end{matrix}$$

Observe que no arquivo texto todos os números estão separados apenas por um espaço e no final da linha existe uma quebra de linha. O código abaixo ilustra como poderia ser realizado o processo de leitura de arquivo, convertendo o conteúdo do arquivo em dados dentro do programa, interpretando-o como matriz:

```
1 >>> A = []
2
3 >>> with open('matriz1.txt', 'r') as file:
4 >>>     for line in file:
5 >>>         line = line.replace('\n', '').split(' ')
6 >>>         A.append([int(x) for x in line])
```

A linha 3 abre o arquivo com nome *'matriz1.txt'* com a opção para leitura (parâmetro *'r'* - read) e em seguida iteramos sobre as linhas do arquivo. Cada linha é interpretada como string e para convertê-lo em inteiro é necessário realizar um tratamento. Na linha 5 do código são retiradas as quebras de linha (*replace*) e o conteúdo da string é separado pelos espaços (*split*). A partir deste momento, *line* é uma lista de strings que contém somente números. Para torná-la em uma lista de inteiros (linha da matriz), na linha 6 iteramos sobre cada elemento realizando um *casting*, convertendo string para inteiros e finalmente anexamos à matriz  $A$ .

## 2 Escrita de Arquivo

Suponha que dentro do programa, existe a matriz  $A$  e desejamos salvá-la em um arquivo com o nome **matriz2.txt**. O código abaixo ilustra como este processo pode ser realizado:

```
1 >>> A = [[17, 24, 1, 15],
2         [23, 5, 7, 16],
3         [ 4, 6, 13, 22],
4         [10, 12, 19, 3],
5         [11, 18, 25, 9]]
6
7 >>> with open('matriz2.txt', 'w') as file:
8 >>>     for line in A:
9 >>>         file.write("_".join(str(x) for x in line))
10 >>>         file.write('\n')
```

De forma análoga à leitura, a linha 7 abre o arquivo com nome *'matriz2.txt'* com a opção para escrita (parâmetro *'w'* - write). Caso exista um arquivo com o mesmo nome, ele será sobrescrito. Iterando sobre as linhas da matriz  $A$ , na linha 9 é realizada a escrita no arquivo, onde iteramos sobre cada elemento da linha, transformando cada número em string, separados por um espaço. Finalmente na linha 10 é inserido a quebra de linha.