

# Programação com Octave/Matlab

Coltec - UFMG

## Capítulo 10 - Entrada e Saída com Arquivos

Márcio Fantini Miranda

16 de junho de 2021

### Sumário

<b>1</b>	<b>Entrada e Saída em Arquivos</b>	<b>1</b>
----------	------------------------------------	----------

## **1 Entrada e Saída em Arquivos**

- O Octave possui recursos importantes que nos auxiliam a salvar e ler dados em arquivos.
- Os dados podem ser salvos em arquivos TEXTOS, isto é que podem ser lidos por um "humano" e...
- ...salvos em arquivos BINÁRIOS, isto é que só podem ser lidos pelas "máquinas" (ou um ser humano muito, muito nerd...)

- Em programação, quando falamos em gravar e ler dados em(de) arquivos, estamos falando em **Entrada e Saída em Arquivos** ou, em Inglês, *file Input/Output*, ou **File I/O**.

As funções do Octave que são usadas para salvar dados em arquivos e para ler dados de arquivos são, respectivamente, **save()** e **load**. Veja os exemplos a seguir.

```
% Exemplo 1 (apenas uma parte)
clear all

a=0;b=1;N=10;
x = 1; y =2; A = [1 2;y -x];z = round(a+b*rand(N,N));
disp(z)

% salva apenas uma variavel:
save a.mat a
% salva três variáveis
save teste.mat a b z
% salva todo o workspace
save tudo.mat
```

- Repare na forma como usamos o comando save.
- Podemos ainda usar o save com várias opções.
- Veja o exemplo 1 completo, no Moodle, parte 11 da aula de Octave.
- Podemos também usar o save() na forma de função, ao invés de usá-lo como um comando

```
save('ab.mat','a','b');  
% ou  
save ('todos.mat');
```

- Existem muitas formas de salvarmos variáveis.
- Podemos salvá-las separadamente ou juntas.
- Vamos usar, na maioria das vezes, o comando `save` se opções.
- Assim salvamos no padrão do Octave, sem nos preocuparmos com o formato do arquivo criado.
- Para lermos o arquivo existente, usamos o comando **load**
- Veja o exemplo:

```
x = -10:0.1:10;  
y = x.^3;  
z = round(2+2*rand(10,10));
```

```
save xy.mat x y;  
clear all  
whos  
load xy.mat  
whos
```

- Existem várias formas de usar as funções (ou comandos) **save()** e **load()**.
- Vamos usar seguindo o padrão (sem acrescentar os tipos *ascii*, *binary* ou *text*).

- Assim o recomendado é usar:

`save NOME-ARQUIVO.mat` (para salvar TODAS as variáveis)

`save NOME-ARQUIVO.mat A B` (para salvar variáveis A e B)

`load NOME-ARQUIVO.mat` (para ler o arquivo)