

# INTRODUÇÃO À LINGUAGEM C E LÓGICA DE PROGRAMAÇÃO – REVISÃO – AULA 4

## Arquivos

### Handling Files

#### 1. Declare FILE type variable

- Use standard library : `#include <stdio.h>`
- ex) `FILE* fp;`

#### 2. Open a file

- Connect FILE type variable with actual file using `fopen` function
- ex) `fp=fopen("test.txt","r");`

#### 3. Perform I/O with the open file

- Textfile : `fscanf()`, `fprintf()`, `fgets()`, `fputs()`, ...
- Binary file : `fread()`, `fwrite()`

#### 4. Close a file

- Break the connection between FILE type variable and actual file
- ex) `fclose(fp);`

Modo	Arquivo	Função
"r"	Texto	Leitura. Arquivo deve existir.
"w"	Texto	Escrita. Cria arquivo se não houver. Apaga o anterior se ele existir.
"a"	Texto	Escrita. Os dados serão adicionados no fim do arquivo ("append").
"rb"	Binário	Leitura. Arquivo deve existir.
"wb"	Binário	Escrita. Cria arquivo se não houver. Apaga o anterior se ele existir.
"ab"	Binário	Escrita. Os dados serão adicionados no fim do arquivo ("append").
"r+"	Texto	Leitura/Escrita. O arquivo deve existir e pode ser modificado.
"w+"	Texto	Leitura/Escrita. Cria arquivo se não houver. Apaga o anterior se ele existir.
"a+"	Texto	Leitura/Escrita. Os dados serão adicionados no fim do arquivo ("append").
"r+b"	Binário	Leitura/Escrita. O arquivo deve existir e pode ser modificado.
"w+b"	Binário	Leitura/Escrita. Cria arquivo se não houver. Apaga o anterior se ele existir.
"a+b"	Binário	Leitura/Escrita. Os dados serão adicionados no fim do arquivo ("append").

# INTRODUÇÃO À LINGUAGEM C E LÓGICA DE PROGRAMAÇÃO – REVISÃO – AULA 4

## Vetores (arrays) e Matrizes (2d arrays)

```
int num[3][4] = {  
    {1, 2, 3, 4},  
    {5, 6, 7, 8},  
    {9, 10, 11, 12}  
};
```

		col →			
		0	1	2	3
row ↓	0	1	2	3	4
	1	5	6	7	8
	2	9	10	11	12

# INTRODUÇÃO À LINGUAGEM C E LÓGICA DE PROGRAMAÇÃO – REVISÃO – AULA 4

## Strings e Arquivos

```
char str[6] = "Hello";
```

index	0	1	2	3	4	5
value	H	e	l	l	o	\0

## String Library Functions

#include <string.h>

Name	Description
<b><i>strlen</i></b>	return the length of string not counting \0
<b><i>strcpy</i></b>	copies string from source to dest
<b><i>strncpy</i></b>	copies n chars from source to dest
<b><i>strcat</i></b>	appends string from source to end of dest
<b><i>strncat</i></b>	appends n chars from source to end of dest
<b><i>strcmp</i></b>	compares two strings alphabetically
<b><i>strncmp</i></b>	compares the first n chars of two strings
<b><i>strstr</i></b>	finds a string inside another
<b><i>strtok</i></b>	breaks string into tokens using delimiters