

## Programação em C/C++

Prof. Márcio Sarroglia Pinho

### Uso de Arquivos Texto

[Criação](#)

[Leitura de Arquivos texto](#)

[Gravação em Arquivos Texto](#)

#### Criação de Arquivos

Para usar as funções de manipulação de arquivo o programador deve incluir a **STDIO.H**

Para usar um arquivo em C é necessário abrí-lo. Para tanto, a linguagem C possui o comando **fopen**.

```
arq = fopen("ArqGrav.txt", "rt");
```

O primeiro parâmetro é o nome do arquivo, o segundo a forma de abertura:

"wt": abertura para gravação, arquivo texto

"rt": abertura para leitura, arquivo texto

A função fopen retorna um "apontador" para o arquivo caso consiga abrí-lo, caso contrário, retorna a constante **NULL**.

#### Exemplo:

```
FILE *arq;
int result;
char Str[50];

arq = fopen("ArqGrav.txt", "rt");

if (arq == NULL)
{
    printf("Problemas na CRIACAO do arquivo\n");
    return;
}
```

#### Arquivos Texto

##### Leitura

Para leitura de arquivos texto pode-se usar a função **fgets** ou **fscanf**.

##### Leitura com FGETS

A função FGETS lê uma linha inteira de uma vez.

Exemplo: `result = fgets(Linha, 100, arq);` // o 'fgets' lê até 99 caracteres ou até o '\n'

Se a função for executada com sucesso, **fgets** retorna o endereço da string lida, caso contrário retorna NULL.

```
// *****
// Exemplo de uso de arquivo texto
// Este programa lê um arquivo texto e imprime o seu
// conteúdo na tela.
// *****
#include <stdio.h>
#include <conio.h>

void main()
{
    FILE *arq;
    char Linha[100];
    char *result;
    int i;

    clrscr();

    // Abre um arquivo TEXTO para LEITURA
    arq = fopen("ArqTeste.txt", "rt");

    if (arq == NULL) // Se houve erro na abertura
    {
        printf("Problemas na abertura do arquivo\n");
        return;
    }

    i = 1;
    while (!feof(arq))
    {
        // Lê uma linha (inclusive com o '\n')
        result = fgets(Linha, 100, arq); // o 'fgets' lê até 99 caracteres ou até o '\n'
        if (result) // Se foi possível ler
            printf("Linha %d : %s", i, Linha);
        i++;
    }
}
```

```
    fclose(arq);
}
```

## Leitura com FSCANF

A função **FSCANF** funciona como a função SCANF, porém, ao invés de ler os dados de teclado, estes dados são lidos de arquivo.

Exemplo:

```
int i, result;
float x;
result = fscanf(arq, "%d%f", &i, &x);
```

Se result for igual à constante EOF, não há mais dados para serem lidos.

---

## Gravação

Para gravação de arquivos texto usa-se as funções [fputs](#) e [fprintf](#).

### Exemplo de fputs:

```
result = fputs(Str, arq);
```

Se a função NÃO for executada com sucesso, **fputs** retorna a constante **EOF**.

```
char Str[100];
FILE *arq;

arq = fopen("ArqGrav.txt", "wt"); // Cria um arquivo texto para gravação
if (arq == NULL) // Se não conseguiu criar
{
    printf("Problemas na CRIACAO do arquivo\n");
    return;
}

strcpy(Str, "Linha de teste");
result = fputs(Str, arq);
if (result == EOF)
    printf("Erro na Gravacao\n");

fclose(arq);
```

### Exemplo de fprintf:

```
result = fprintf(arq, "Linha %d\n", i);
```

Se a função **fprintf** for executada com sucesso, devolve o número de caracteres gravados. Se a função NÃO for executada com sucesso, retorna a constante **EOF**.

```
// *****
// Exemplo de uso de arquivo texto
// Este programa grava um arquivo texto
// *****
#include <stdio.h>
#include <conio.h>

void main()
{
    FILE *arq;
    int i;
    int result;

    clrscr();
    arq = fopen("ArqGrav.txt", "wt"); // Cria um arquivo texto para gravação

    if (arq == NULL) // Se não conseguiu criar
    {
        printf("Problemas na CRIACAO do arquivo\n");
        return;
    }

    for (i = 0; i < 10; i++)
    {
        // A funcao 'fprintf' devolve o número de bytes gravados
        // ou EOF se houve erro na gravação
        result = fprintf(arq, "Linha %d\n", i);
        if (result == EOF)
            printf("Erro na Gravacao\n");
    }
    fclose(arq);
}
```