3. Linguagem de Programação C

Prof. Renato Tinós

Departamento de Computação e Matemática
(FFCLRP/USP)

Principais Tópicos

3.9. Arquivos

- 3.9.1. Introdução
- 3.9.2. Escrita
- 3.9.3. Leitura
- 3.9.4. Exercícios

3.9.1. Introdução

- **Um arquivo em C pode representar**
  - Arquivo em disco
  - Um teclado
  - Uma impressora
  - Outros dispositivos de Entrada/Saída (E/S)

- **Aqui, só os arquivos em disco serão considerados**

3.9.1. Introdução

- **Biblioteca *stdio.h***

  – Fornece várias funções para a manipulação de arquivos

  – Possui um tipo de dados para ser utilizado com arquivos

  FILE

3.9.1. Introdução

- **A linha comum que une o sistema de E/S é o *ponteiro de arquivo***
  - É um ponteiro para as informações de vários dados sobre um arquivo
  - Identifica um arquivo específico em disco
  - É usado pelo fluxo associado a ele para direcionar a operação das funções de E/S
  - É uma variável do tipo ***FILE***

3.9.1. Introdução

- **Variável do tipo *FILE***
  - É capaz de identificar um arquivo no disco, direcionando-lhe todas as operações
  - São declaradas como

*FILE *nome_do_ponteiro_de_arquivo*

3.9.1. Introdução

- **Abrindo Arquivos: função *fopen( )***
  - Abre um fluxo e o liga a um arquivo
  - Retorna o ponteiro de arquivo associado àquele arquivo

*FILE *fopen(char *variavel_nome_arquivo , char *variavel_modo)*

ou

*FILE *fopen("**nome do arquivo**" , "**modo**")*

3.9.1. Introdução

- **Modos de abertura: arquivo em <u>formato texto</u>**

"rt" : abre um arquivo para leitura (arquivo deve existir)
"wt" : cria um arquivo para escrita (deleta e cria um novo arquivo se ele já existir)
"at" : anexa dados em um arquivo já existente (cria arquivo se ele não existir)
"rt+" : abre um arquivo para leitura e escrita (arquivo deve existir)
"wt+" : cria um arquivo para leitura e escrita (deleta e cria um novo arquivo se ele já existir)
"at+" : anexa dados em um arquivo já existente, permitindo também a leitura (cria arquivo se ele não existir)

  - Operações começam:
    - Início do arquivo: rt, wt, rt+, wt+
    - Fim do arquivo: at, at+

3.9.1. Introdução

- **Modos de abertura: arquivo em formato binário**

"rb" : abre um arquivo para leitura (arquivo deve existir)
"wb" : cria um arquivo para escrita (deleta e cria um novo arquivo se ele já existir)
"ab" : anexa dados em um arquivo já existente (cria arquivo se ele não existir)
"rb+" : abre um arquivo para leitura e escrita (arquivo deve existir)
"wb+" : cria um arquivo para leitura e escrita (deleta e cria um novo arquivo se ele já existir)
"ab+" : anexa dados em um arquivo já existente, permitindo também a leitura (cria arquivo se ele não existir)

  - Operações começam:
    - Início do arquivo: rb, wb, rb+, wb+
    - Fim do arquivo: ab, ab+

3.9.1. Introdução

- **A função *fopen( )* retorna**
  - Um ponteiro de arquivo que não deve ter seu valor alterado pelo programa
  - Um nulo se ocorre erro na abertura do arquivo

3.9.1. Introdução

- **Exemplos**

```
...
FILE *arquivo
arquivo = fopen("nome_do_arquivo.dat", "w");
...
```

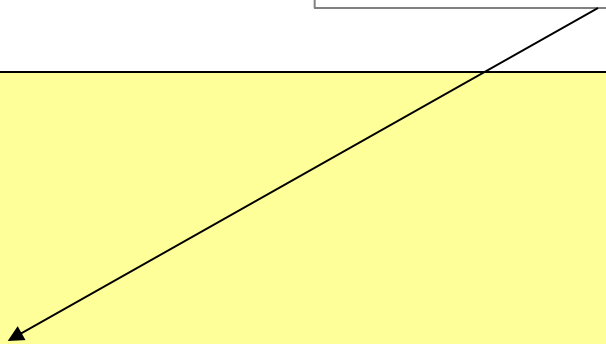
```
...
FILE *arquivo
arquivo = fopen("nome_do_arquivo.dat", "w");
if ( arquivo==NULL) {
        printf("O arquivo nao pode ser aberto \n");
        exit(1);
}
...
```

Sai do programa, indicando que um erro ocorreu

3.9.1. Introdução

- **Exemplos**

Diretório deve ser válido!



```
# include <stdio.h>

main () {
        FILE *arq;
        arq = fopen("\tmp\arquivo_teste.txt", "wb");

        if(arq != NULL)
          printf("Arquivo criado com sucesso!");
        else
          printf("Falha ao abrir o arquivo");

}
```

3.9.1. Introdução

- **Fechando arquivos: função *fclose( )***

  – Se um erro ocorre no programa enquanto um arquivo está aberto, pode ocorrer
    - perda de dados
    - perda do arquivo

  – Para evitar tais problemas, a função *fclose( )* é utilizada para fechar um arquivo

```
int fclose(FILE *variavel_ponteiro_de_arquivo);
```

3.9.1. Introdução

- **A função *fclose( )* gera como resultado um inteiro**
  - Quando o resultado for igual a zero, significa que o arquivo foi fechado corretamente
  - Quando o resultado é diferente de zero, significa que ocorreu um erro ao fechar o arquivo

3.9.2. Escrita

- **Escrita de dados em arquivo**
  - Função ***fprintf()***
    - Similar à função printf()

```
        ...
        fprintf(arq, “Informações\n”);
        fprintf(arq, “Valor: %d \n”, v);
        fprintf(arq, “Nota: %1.2f \n”, nota);
...
```

3.9.3. Leitura

- **Leituras de dados de arquivo**
  - Função ***feof()***
    - Verifica se o final do arquivo foi encontrado.

```
...
while (feof(arq) == 0) {
        // leitura dos dados
}
...
```

3.9.3. Leitura

- **Leituras de dados de arquivo**
  - Função ***fscanf()***
    - Similar à função scanf()

```
...
int v;
char nome[50];
while (feof(arq) == 0) {
        fscanf(arq, “%d”, &v);
        fscanf(arq, “%s”, nome);
}
...
```

3.9.3. Leitura

- **Detectando erros: função *ferror( )***

  – Utilizada para determinar se uma operação em um arquivo produziu um erro

  – Se ocorre um erro na leitura ou escrita de um arquivo tipo texto, a função de E/S retorna **EOF**
    - A função *ferror()* é utilizada para verificar o que ocorreu

```
int ferror(FILE *variavel_ponteiro_de_arquivo);
```

3.9.3. Leitura

- **A função *ferror( )***

  – Retorna verdadeiro se um erro ocorreu durante a última operação com o arquivo (caso contrário, retorna falso)

  – Uma vez que cada operação determina uma condição de erro, a função deve ser chamada imediatamente após cada operação com o arquivo

3.9.3. Exercícios

Ex. 3.9.1 Escrever um programa em C que permita:

- Cadastrar livros (com nome, autor e editora)
- Ver informações cadastradas (de todos os livros)
- Salvar em um arquivo as informações cadastradas

3.9.3. Exercícios

Ex. 3.9.2 Escrever um programa em C que:

- Leia uma matriz de números inteiros digitados pelo usuário
- Ache o maior elemento de cada linha
- Salvar em um arquivo os maiores elementos de cada linha (utilizando arquivos do tipo *at*)
- Imprima na tela os dados armazenados.

3.9.3. Exercícios

Ex. 3.9.3 Escrever um programa em C que:

- Armazene valores do tipo inteiro em um vetor de 10 posições
- Salve em um arquivo os números que são ímpares
- Leia tais números do arquivo