

Algoritmos de Programação 2

Professor: José Eurípedes Ferreira de Jesus Filho
jeferreirajf@gmail.com

Universidade Federal de Jataí – UFJ

Hoje vamos aprender a salvar nossos dados!



Introdução

- Até agora, todos **os dados** que lidamos são perdidos ao fechar o programa.
- Isso acontece porque eles estão na **memória volátil (RAM)** do computador.
- É comum que um programa quando reaberto continue de onde parou.
- Hoje iremos aprender a manipular arquivos de texto!



O que são arquivos

- Arquivos são uma sequência de bytes na **memória lenta** (ROM).
- Os bytes de um arquivo **não possuem endereços**.
- Para acessar o 5º byte de um arquivo **é necessário passar** pelo 1º, ..., 4º bytes.



Tipos de arquivos

- Arquivos de texto.

- ✓ São arquivos comuns que podem ser criados por qualquer editor de texto (.txt).
- ✓ Armazenam os dados no formato texto (ASCII geralmente).
- ✓ Ao abrir um arquivo de texto, o conteúdo pode ser facilmente lido e editado.
- ✓ Não são seguros.
- ✓ Geralmente ocupam mais espaço do que arquivos binários.

- Arquivos binários.

- ✓ São os arquivos .bin .
- ✓ Guardam os dados em formato binário (0 ou 1).
- ✓ Não são lidos facilmente.
- ✓ Maior segurança.



O tipo FILE

- Em C, FILE é um tipo presente na biblioteca `stdio.h`.
- Quando se quer operar com arquivos é necessário associar o arquivo com uma variável do tipo FILE.
- Esta variável fará a comunicação entre o arquivo e o programa.

```
FILE *file;
```



Operando com arquivos de texto.

- Para conseguir operar com arquivos de texto é necessário o seguinte:

1. **Criar uma variável do tipo FILE.**

```
FILE *file;
```

2. **Associar o arquivo com a variável.**

```
file = fopen(ENDEREÇO_ARQUIVO, MODO_ABERTURA);
```

3. **Executar as devidas operações.**

```
fprintf(file, "impressao", VARIAVEIS);  
fscanf(file, FORMATO, &VARIAVEIS);  
fgetchar(), fputc(), ... Etc
```

4. **Fechar o arquivo.**

```
fclose(file);
```



Operando com arquivos de texto.

- A função `fopen()` vincula o arquivo desejado com uma variável do tipo `FILE`.
- A função retorna ou uma variável `FILE` ou `NULL`, caso não consiga abrir o arquivo desejado.

```
FILE *file;  
file = fopen("arquivo.txt", r);
```



Operando com arquivos de texto.

- É necessário informar o modo de operação que se deseja abrir o arquivo.
 - ✓ r para leitura.
 - ✓ w para escrita em sobreposição.
 - ✓ a para adição de informação.
 - ✓ r+ abre para leitura e escrita em sobreposição.
 - ✓ w+ abre para leitura e escrita em sobreposição.
 - ✓ a+ abre para leitura e adição de informação.



Operando com arquivos de texto.

- As funções `fprintf()` e `fscanf()` funcionam de forma muito parecida com `printf()` e `scanf()`.
- Escrevendo um inteiro de um arquivo:

```
FILE *file;  
file = fopen("arquivo.txt", w);  
int var = -1;  
fprintf(file, "%d", var);
```



Operando com arquivos de texto.

- As funções `fprintf()` e `fscanf()` funcionam de forma muito parecida com `printf()` e `scanf()`.
- Lendo um inteiro de um arquivo:

```
FILE *file;  
file = fopen("arquivo.txt", r);  
int var;  
fscanf(file, "%d", &var);
```



Operando com arquivos de texto.

- A função `fscanf()` retorna a quantidade de elementos lidos.
- Isso pode ser utilizado para detectar o final do arquivo.

```
FILE *file;  
file = fopen("arquivo.txt", r);  
int var;  
while( fscanf(file, "%d", &var) > 0 )  
    printf("Numero lido: %d", var);
```



Exercícios

- Desenvolva um programa que leia todos os inteiros de um arquivo de texto e imprima na tela a média dos valores lidos.



Operando com arquivos de texto.

- A função `fclose()` desvincula a variável do tipo `FILE` com o arquivo.

```
FILE *file;  
file = fopen("arquivo.txt", r);  
int var;  
fscanf(file, "%d", &var);  
fclose(file);
```

- É **MUITO IMPORTANTE** sempre fechar o arquivo após operar com ele.



Informações adicionais

- Geralmente, os sistemas operacionais permitem infinitos acessos a um único arquivo desde que esses acessos sejam para leitura.
- Somente **UM ÚNICO** acesso é permitido caso o arquivo seja aberto para escrita.
- É uma boa prática de programação e de design de software sempre abrir uma conexão com um arquivo utilizando o modo correto.



Resumo

- **ARQUIVOS**

- ✓ Sequência de bytes na memória não volátil.
- ✓ O acesso aos bytes de um arquivo é feito de maneira sequencial.

- **TIPOS DE ARQUIVOS**

- ✓ Texto.
- ✓ Binário.

- **OPERAÇÕES COM ARQUIVOS DE TEXTO**

- ✓ Necessário uma variável do tipo FILE para a comunicação entre o programa e o arquivo.
- ✓ Função `fopen()` para fazer a associação.
- ✓ Funções `fprintf()`, `fscanf()`, `fputc()`, `fgetchar()`, etc, para fazer operações nos arquivos.
- ✓ Função `fclose()` para desfazer a associação.

