

## DISCIPLINA: ESTRUTURAS DE DADOS

## PROGRAMA DE ARTICULAÇÃO DA FORMAÇÃO PROFISSIONAL MÉDIA E SUPERIOR (AMS)

## ARQUIVOS BINÁRIOS

- Prof. Alexandre Ponce de Oliveira  
alexandreponceo@gmail.com

Aulas 21 e 22

Lins/SP

# Sistema de Arquivos – Arquivos Binários

---

É indicado para manipular *structs* (registros)

A gravação em formato binário ocupa menos bytes que arquivos textos.

Permite ler e escrever do arquivo binário considerando toda a *struct* e não apenas cada elemento que a compõe.

É necessária a biblioteca `<stdlib.h>`

# Sistema de Arquivos – Arquivos Binários

---

## **Funções para leitura e gravação em arquivos binários**

As funções `fread()` e `fwrite()` são utilizadas para leitura e escrita de dados em modo binário, conforme as seguintes sintaxes:

`fread (variável, tam_bytes, qtde, arquivo)`

`fwrite (variável, tam_bytes, qtde, arquivo)`

- `tam_bytes` é um valor que define o número de bytes da estrutura que deve ser lida/escrita.
- `qtde` é o número de estruturas que devem ser lidas/escritas.

# Sistema de Arquivos – Arquivos Binários

---

## Exemplos:

```
FILE *arq;  
int i[5] = {1,2,3,4,5}, cont=0;  
if((arq = fopen("num1.dat", "wb")) == NULL)  
    printf("Erro ao abrir o arquivo copia!!!\n");  
fwrite (&i, sizeof(int), 5, arq);  
fclose(arq);
```

```
if((arq = fopen("num1.dat", "rb")) == NULL)  
    printf("Erro ao abrir o arquivo copia!!!\n");  
fread (&i, sizeof(int), 5, arq);  
for (cont=0; cont<5; cont++)  
{  
    printf ("%d ", i[cont]); }  
fclose(arq);  
getch();
```

A função `sizeof()` retorna a quantidade de bytes de um determinado tipo ou variável.

# Sistema de Arquivos – Arquivos Binários

---

Posicionar o cursor em uma determinada posição do arquivo:

```
int fseek (FILE *arquivo ,int numbytes, int modo);
```

## Parâmetros

Deslocamento: quantidade (em bytes) que deseja-se deslocar para o posicionamento do cursor (é possível ser negativo).

Modo: forma do deslocamento, descritas a seguir:

SEEK_SET	-	posição inicial do arquivo
SEEK_CUR	-	posição corrente no arquivo
SEEK_END	-	posição final do arquivo

Caso seja retornado 0, foi possível fazer o deslocamento.

# Sistema de Arquivos – Arquivos Binários

---

Posicionar o cursor em uma determinada posição do arquivo:

Exemplos:

Final do arquivo:

```
fseek (arq, 0, SEEK_END);
```

Posição específica:

```
scanf ("%d", &posicao);  
fseek (arq, (posicao-1) * sizeof(struct notas), SEEK_SET);
```

# Exemplo1 – Programa gravar em um arquivo binário

```
#include <stdio.h> #include <stdlib.h>
#include <string.h> #include <conio.h>
struct notas {
    char nome[30];
    float media;
};
int main() {
    FILE *arq;
    struct notas aluno[10];
    int i,qtde=2;
    arq = fopen("aula07bin.dat", "wb"); // Abrir arquivo
    for (i=0;i<qtde;i++)
    {
        printf("\nDigite o nome do aluno: ");
        scanf("%s", &aluno[i].nome);
        printf("Digite a media: ");
        scanf("%f", &aluno[i].media);
        fwrite (&aluno[i], sizeof(struct notas), 1, arq);
    }
    fclose(arq);    // Fechar o arquivo
}
```

## Exemplo2 – Programa ler um arquivo binário

```
#include <stdio.h>   #include <stdlib.h>
#include <string.h>  #include <conio.h>
struct notas {
    char nome[30];
    float media;
};
int main() {
    FILE *arq;
    struct notas aluno[10];
    int i,qtde=2;
    arq = fopen("aula07bin.dat", "rb"); // Abrir arquivo
    for (i=0; i<qtde; i++)
    {
        fread (&aluno[i], sizeof(struct notas), 1, arq);
        printf ("Nome[%d]: %s ", i,aluno[i].nome);
        printf ("Media: %f \n", aluno[i].media);
    }
    fclose(arq);    // Fechar o arquivo
    getch();
}
```



## Exemplo3 – Programa ler uma posição específica

```
#include <stdio.h>   #include <stdlib.h>
#include <string.h>   #include <conio.h>
struct notas {
    char nome[30];
    float media;
};
int main() {
    FILE *arq;
    struct notas aluno[10];
    int posicao;
    arq = fopen("aula07bin.dat", "rb"); // Abrir o arquivo
    printf("Digite a posição do registro: ");
    scanf("%d",&posicao);
    fseek (arq, (posicao-1) * sizeof(struct notas), SEEK_SET);
    fread (&aluno[posicao], sizeof(struct notas), 1, arq);
    printf ("Nome: %s ", aluno[posicao].nome);
    printf ("Media: %f \n", aluno[posicao].media);
    fclose(arq);    // Fechar o arquivo
    getch();
}
```

# Exercícios

---

- 1) Escreva um programa para armazenar (arquivo binário) os dados de 5 pacientes com a seguinte estrutura:

```
struct paciente{  
    char nome[50];  
    int idade;  
    float altura;  
    float peso;  
    float imc;  
};
```

- 2) Escreva um programa para mostrar os pacientes cadastrados do exercício anterior.
- 3) Escreva um programa para mostrar apenas o terceiro paciente cadastrado do primeiro exercício.

# Exercícios

---

- 4) Faça um programa que leia uma quantidade de números determinada pelo usuário e armazene em um arquivo binário de inteiros, chamado numeros.bin. Na sequência crie um outro arquivo binário de inteiros, chamado pares.bin, e grave apenas os números pares do arquivo numeros.bin.
- 5) Faça um programa para manipular dados em um arquivo binário com as seguintes funcionalidades inserir, alterar, excluir, listar e sair. Crie um menu de opções para o usuário.