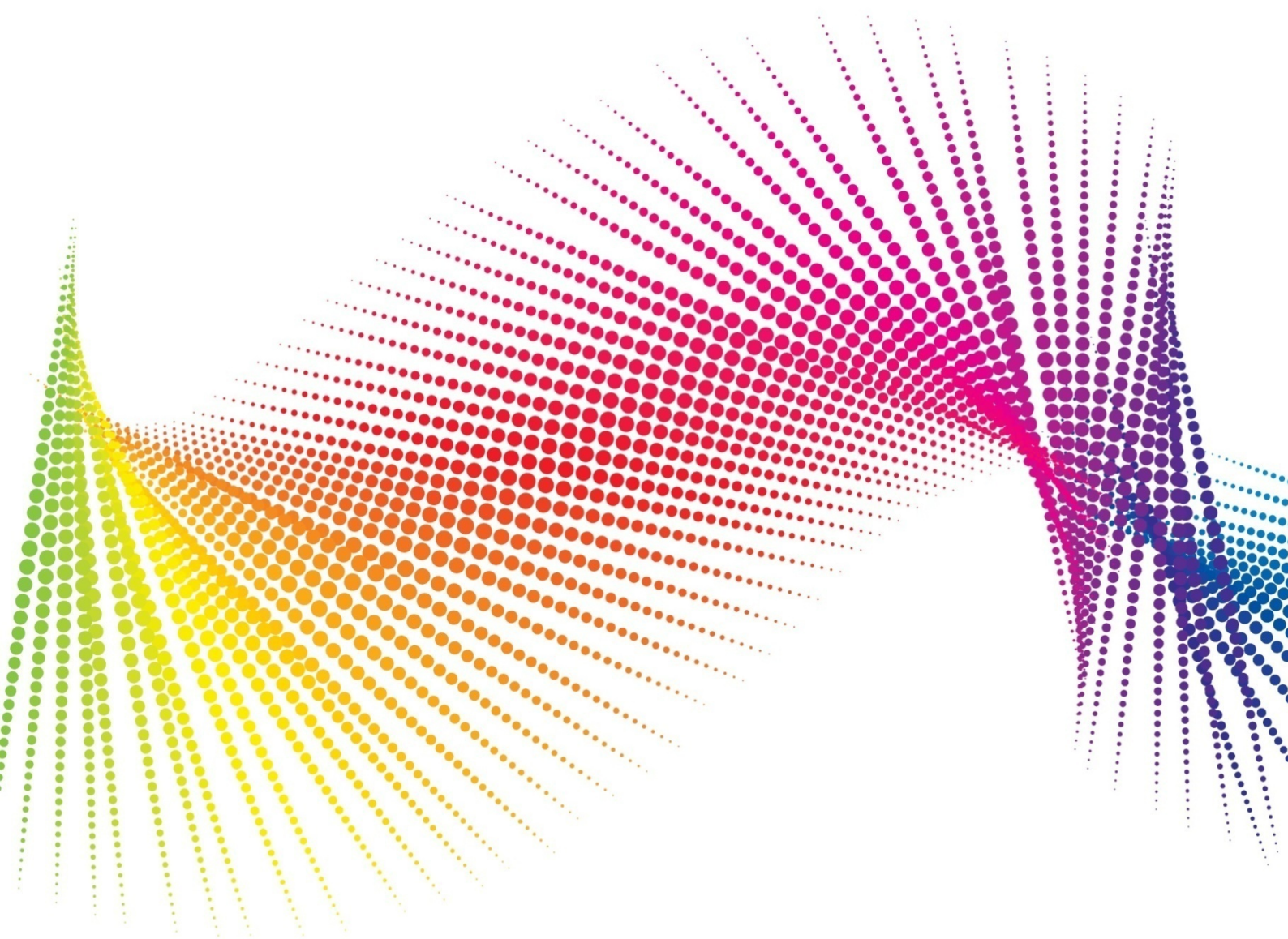


Estrutura de Dados

Aula 17



Este material é parte integrante da disciplina oferecida pela UNINOVE.

O acesso às atividades, conteúdos multimídia e interativo, encontros virtuais, fóruns de discussão e a comunicação com o professor devem ser feitos diretamente no ambiente virtual de aprendizagem UNINOVE.

Uso consciente do papel.

Cause boa impressão, imprima menos.

Aula 17: Arquivos binários

Objetivo: Continuar estudando os arquivos, mostrando como criar e manipular arquivos binários.

Caro aluno, antes de iniciarmos o estudo de arquivos binários, vamos precisar conhecer as funções `fwrite()` e `fread()`.

A função `fwrite()`

A função `fwrite()` permite a gravação de um conjunto de bytes de dados num arquivo. Vejamos as características da função `fwrite()`:

- Arquivo cabeçalho a ser incluído: **<stdio.h>**.
- Permite a escrita, em um arquivo, de dados maiores que 1 byte, para qualquer tipo de dado.
 - Recebe como argumento um ponteiro para as informações que serão escritas no arquivo, o número de bytes e de itens a serem escritos, cada um com o número de bytes especificado e um ponteiro para arquivo.
 - Devolve o número de itens escritos, que será igual ao número especificado, a não ser na ocorrência de um erro.

A função `fread()`

A função `fread()` permite a leitura de um conjunto de bytes de dados de um arquivo. Vejamos as características da função `fread()`:

- Arquivo cabeçalho a ser incluído: **stdio.h**.
- Permite a leitura em arquivo de dados maiores que 1 byte para qualquer tipo de dado.

- Recebe como argumento um ponteiro para uma região de memória que receberá os dados do arquivo, o número de bytes e de itens a serem lidos, cada um com o número de bytes especificado, e um ponteiro para arquivo.
- Devolve o número de itens lidos, o qual pode ser menor que o número especificado, se o final do arquivo for atingido ou na ocorrência de um erro.

Exemplo de um programa que trabalha com arquivo binário, struct e usa as funções fread() e fwrite()

Caro aluno, este programa tem uma intenção bem didática, então vamos inserir comentários ao longo do código explicando o que está sendo feito, ok? Preste bastante atenção para que consiga entender bem.

```
/* arquivoBinario.c: Ilustra a criacao/leitura de um arquivo binario no formato
                        nomedoaluno mediafinal curso
e a impressao na tela dos dados gravados no arquivo e o nome completo do curso a partir
de sua sigla. Utiliza as funcoes fread() fwrite() feof() e struct
*/

#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<string.h>

//define a função que limpa a tela como limpatela();
#define limpatela(); system("cls");

// cor de fundo preta e letras azuis brilhantes
#define corpretoeazul(); system ("color 0B");

// cor de fundo azul e letras azuis claras
#define corazuleamarelo(); system ("color 1E");

//define a estrutura que vai ser gravada e lida do arquivo
struct a {
    char nome[50];
    float media;
    char curso[5];
};

struct a aluno;
```

// o usuário é que vai escolher o nome do arquivo

```
char nomearq[50];
//ponteiro de arquivo arq (nome logico do arquivo)
FILE *arq;
```

//conta o número de dados gravados no arquivo binário

```
int cont=0;
```

```
int main()
```

```
{
```

```
    limpatela();
    corazuleamarelo();
```

```
    puts("*****");
    puts("\nCria ou abre e insere dados num arquivo binario");
    puts(" depois mostra os dados e o nome do curso");
```

```
    printf("\nNome do arquivo binario a ser criado ou aberto (com
extensao): ");
    scanf("%s",nomearq);
```

/*O modo de abertura a+b indica que se trata de um arquivo binário. Se o arquivo já existir ele será aberto; se não existir, será criado */

```
arq = fopen(nomearq,"a+b");
```

//se fopen retornar NULL o arquivo não poderá nem ser aberto nem criado

```
if (arq==NULL)
{ printf("\nProblemas na abertura ou criacao do arquivo %s.",
        nomearq);
  printf("\nO programa sera finalizado.");
  getch();
  return 0; //finaliza o programa
}
```

//aqui o arquivo foi aberto ou criado com sucesso

```
printf("\nCriando ou inserindo dados no arquivo binario %s",
        nomearq);
```

```
cont=0;
```

```
while (1)
```

```
{
    //neste trecho os dados serão fornecidos pelo usuário,
    //armazenados numa estrutura e gravados no arquivo binário
    //usando a função fwrite()
    fflush(stdin);
    printf("\nNome do aluno . finaliza: ");
    gets(aluno.nome);
```

```

        if (strcmp(aluno.nome, ".")==0)
            break;

        printf("\nMedia de notas: ");
        scanf("%f",&aluno.media);
        fflush(stdin);
        printf("\nCurso (SI, CC ou TADS): ");
        gets(aluno.curso);
        strupr(aluno.curso);

        fwrite(&aluno,sizeof(aluno),1,arq);

        fflush(stdin);
    }
//todos os dados fornecidos foram gravados. Fecha o arquivo
fclose(arq);

/*Abre o arquivo novamente.
O modo de abertura a+b indica que se trata de um arquivo binário.
Neste caso o arquivo já existe e será aberto. */

arq = fopen(nomearq,"a+b");
if (arq==NULL)
{ printf("\nProblemas na abertura do arquivo %s.", nomearq);
  printf("\nO programa sera finalizado.");
  getch();
  return 0; //finaliza o programa
}

while (! feof(arq))
{
//neste trecho os dados gravados no arquivo binário serão lidos
//usando a função fread(), transferidos para a estrutura e
//impressos na tela e contados

    fread(&aluno,sizeof(aluno),1,arq);
    printf("\nNome do aluno: %s", aluno.nome);
    printf("\nMedia de notas: %.1f", aluno.media);
    printf("\nCurso: %s", aluno.curso);
    ++cont;

//a sigla do curso será impressa por completo
    if (strcmp( aluno.curso, "SI")==0)
        printf("-Sistemas de Informacao\n");
    if (strcmp( aluno.curso, "CC")==0)
        printf("-Ciencia da Computacao\n");
    if (strcmp( aluno.curso, "TADS")==0)
        printf("-Tecnologia em Analise de Sistemas\n");

```

```
        printf("\n");
        getch();
    }
    fclose(arq);
    //todos os dados foram lidos e impressos na tela. Fecha o arquivo
    //imprime a quantidade de dados gravados no arquivo
    if (cont==0)
        printf("\nArquivo %s vazio ou inexistente",nomearq);
    else
        printf("\nArquivo %s possui %i alunos registrados\n",
            nomearq, cont);
    puts("*****");

    //finaliza o programa
    printf("\nFim do programa");
    getch();
    return 0;
}
```

Caro aluno, certamente agora que você entendeu bem este programa que mostra como criar, gravar e ler dados de um arquivo binário, você vai implementar e testar este programa. Depois vá ao diretório corrente e verifique se é possível visualizar os dados gravados no arquivo binário criado. Outro exercício indicado é executar o programa novamente e acrescentar mais dados ao mesmo arquivo. Você verá que os dados anteriores serão mantidos e os novos serão acrescentados.

Agora, caro aluno, vamos praticar resolvendo os exercícios propostos. **Leia a lista, resolva os exercícios e verifique seu conhecimento.** Caso fique alguma dúvida, leve a questão ao Fórum e divida com seus colegas e professor.

REFERÊNCIAS

SCHILDT, H. *C Completo e total*. São Paulo: Makron Books, 1997.

TENEMBAUM, Aaron M., et al. *Estruturas de dados usando C*. São Paulo: Pearson Makron Books, 1995.