

Algoritmos e Estruturas de Dados II

Aula 17 – Arquivos: Exemplos de Códigos

Prof. Luciano A. Digiampietri
digiampietri@usp.br
@digiampietri

2025

Gerenciamento Básico de Arquivos

Implementaremos algumas funções básicas para:

Gerenciamento Básico de Arquivos

Implementaremos algumas funções básicas para:

- Gravação e Leitura de Arquivo Texto

Gerenciamento Básico de Arquivos

Implementaremos algumas funções básicas para:

- Gravação e Leitura de Arquivo Texto
- Gravação e Leitura de Arquivo Binário

Gerenciamento Básico de Arquivos

Implementaremos algumas funções básicas para:

- Gravação e Leitura de Arquivo Texto
- Gravação e Leitura de Arquivo Binário
- Gravação e Leitura de um Grafo (arquivo texto)

Gerenciamento Básico de Arquivos

Implementaremos algumas funções básicas para:

- Gravação e Leitura de Arquivo Texto
- Gravação e Leitura de Arquivo Binário
- Gravação e Leitura de um Grafo (arquivo texto)
- Gravação de um Grafo (arquivo binário)

Gerenciamento Básico de Arquivos

Implementaremos algumas funções básicas para:

- Gravação e Leitura de Arquivo Texto
- Gravação e Leitura de Arquivo Binário
- Gravação e Leitura de um Grafo (arquivo texto)
- Gravação de um Grafo (arquivo binário)
- Gravação e Leitura de Registros de Tamanho Fixo

Gerenciamento Básico de Arquivos

Implementaremos algumas funções básicas para:

- Gravação e Leitura de Arquivo Texto
- Gravação e Leitura de Arquivo Binário
- Gravação e Leitura de um Grafo (arquivo texto)
- Gravação de um Grafo (arquivo binário)
- Gravação e Leitura de Registros de Tamanho Fixo
- Gravação e Leitura de Registros de Tamanho Variável

Gravação em Arquivo Texto - `fprintf`

```
void salvarEmArquivoTexto(char* nomeDoArquivo,
                          char* texto){
    FILE *arq;
    if (!(arq = fopen(nomeDoArquivo, "w"))){
        printf("Nao foi pos...: '%s'\n", nomeDoArquivo);
        exit(1);
    }
    fprintf(arq, "%s", texto);
    fclose(arq);
}
```

Gravação em Arquivo Texto - fprintf

```
void adicionarEmArquivoTexto(char* nomeDoArquivo,
                             char* texto){
    FILE *arq;
    if (!(arq = fopen(nomeDoArquivo, "a"))){
        printf("Nao foi pos...: '%s'\n", nomeDoArquivo);
        exit(1);
    }
    fprintf(arq, "%s", texto);
    fclose(arq);
}
```

Leitura de Arquivo Texto - fscanf

```
void lerArquivoTexto(char* nomeDoArquivo){
    FILE *arq;
    char c;
    if (!(arq = fopen(nomeDoArquivo, "r"))){
        printf("Nao foi pos...: '%s'\n", nomeDoArquivo);
        exit(1);
    }
    printf("\nExibindo ...: '%s'\n", nomeDoArquivo);
    while (fscanf(arq,"%c",&c) != EOF){
        printf("%c", c);
    }
    fclose(arq);
}
```

Arquivo Texto - Primeiro Exemplo

```
#include <stdio.h>
#include <stdlib.h>
...
int main(){
    salvarEmArquivoTexto("Exemplo.txt", "Prezados(as)
        alunos(as)\nNão esqueçam de estudar para as provas,
        fazer os trabalhos e beber água.\n
        Atenciosamente,\nDigi\n");
    lerArquivoTexto("Exemplo.txt");
    printf("\n");
    adicionarEmArquivoTexto("Exemplo.txt", "\nPS.: BOA
        NOITE!!!\n");
    lerArquivoTexto("Exemplo.txt");
    return 0;
}
```

Arquivo Texto - Primeiro Exemplo

```
Exibindo conteudo do arquivo: 'Exemplo.txt'  
Prezados(as) alunos(as)  
Não esqueçam de estudar para as provas, fazer os  
trabalhos e beber água.  
Atenciosamente,  
Digi
```

```
Exibindo conteudo do arquivo: 'Exemplo.txt'  
Prezados(as) alunos(as)  
Não esqueçam de estudar para as provas, fazer os  
trabalhos e beber água.  
Atenciosamente,  
Digi
```

```
PS.: BOA NOITE!!!
```

Gravação em Arquivo Texto - `fwrite`

```
void salvarEmArquivo(char* nomeDoArquivo,
                    char* texto, int n){
    FILE *arq;
    if (!(arq = fopen(nomeDoArquivo, "w"))){
        printf("Nao foi pos...: '%s'\n", nomeDoArquivo);
        exit(1);
    }
    fwrite(texto, 1, n, arq);
    fclose(arq);
}
```

Gravação em Arquivo Texto - `fwrite`

```
void adicionarEmArquivo(char* nomeDoArquivo,
                        char* texto, int n){
    FILE *arq;
    if (!(arq = fopen(nomeDoArquivo, "a"))){
        printf("Nao foi pos...: '%s'\n", nomeDoArquivo);
        exit(1);
    }
    fwrite(texto, 1, n, arq);
    fclose(arq);
}
```

Leitura de Arquivo Texto - fgetc

```
void lerArquivoTexto(char* nomeDoArquivo){
    FILE *arq;
    char c;
    if (!(arq = fopen(nomeDoArquivo, "r"))){
        printf("Nao foi pos...: '%s'\n", nomeDoArquivo);
        exit(1);
    }
    printf("\nExibindo ....: '%s'\n", nomeDoArquivo);
    while (!feof(arq)){
        c = fgetc(arq);
        if (c==-1) break;
        printf("%c", c);
    }
    fclose(arq);
}
```

Arquivo Texto - Segundo Exemplo

```
#include <stdio.h>
#include <stdlib.h>
...
int main(){
    salvarEmArquivo("Exemplo2.txt", "Prezados(as) alunos
        (as)\nNão esqueçam de estudar para as provas,
        fazer os trabalhos e beber água.\nAtenciosamente,
        \nDigi\n", 121);
    lerArquivo("Exemplo2.txt");
    printf("\n");
    adicionarEmArquivo("Exemplo2.txt", "\nPS.: BOA NOITE!
        !!\n",19);
    lerArquivo("Exemplo2.txt");
    return 0;
}
```

Arquivo Texto - Primeiro Exemplo

```
Exibindo conteudo do arquivo: 'Exemplo2.txt'  
Prezados(as) alunos(as)  
Não esqueçam de estudar para as provas, fazer os  
trabalhos e beber água.  
Atenciosamente,  
Digi
```

```
Exibindo conteudo do arquivo: 'Exemplo2.txt'  
Prezados(as) alunos(as)  
Não esqueçam de estudar para as provas, fazer os  
trabalhos e beber água.  
Atenciosamente,  
Digi
```

```
PS.: BOA NOITE!!!
```

Gravação de Grafo em Arquivo

```
void salvaGrafo(Grafo* g, char* nomeDoArquivo){
    if (!g) return;
    FILE *arq;
    if (!(arq = fopen(nomeDoArquivo, "w"))){
        printf("Nao foi pos...: '%s'\n", nomeDoArquivo);
        exit(1);
    }
    int x, y;
    fprintf(arq, "%i\n", g->numVertices);
    for (x=0;x<g->numVertices;x++){
        for (y=0;y<g->numVertices;y++){
            fprintf(arq,"%i ",g->matriz[x][y]);
            fprintf(arq,"\n");
        }
    }
    fclose(arq);
}
```

Leitura de Grafo de um Arquivo

```
Grafo* leGrafo(char* nomeDoArquivo){
    FILE *arq;
    if (!(arq = fopen(nomeDoArquivo, "r"))){
        printf("Nao foi pos...: '%s'\n", nomeDoArquivo);
        exit(1);
    }
    int v, arestas = 0;
    bool a;
    int y, x=0;
    fscanf(arq, "%i", &v);
    printf("\nCarregando grafo com %i vertices.\n", v);
    ...
}
```

Leitura de Grafo de um Arquivo

```
...
Grafo* g = (Grafo*) malloc(sizeof(Grafo));
inicializaGrafo(g, v);
for (x=0;x<v;x++)
    for (y=0;y<v;y++){
        fscanf(arq, "%i", &a);
        if (a){
            arestas++;
            g->matriz[x][y] = a;
        }
    }
g->numArestas = arestas;
fclose(arq);
return g;
}
```

Grafo - Arquivo Texto

```
int main(){
    Grafo* g = criaGrafoAleatorio(5,6);
    exhibeGrafo(g);
    salvaGrafo(g, "digrafo.txt");

    Grafo* g2 = leGrafo("digrafo.txt");
    exhibeGrafo(g2);
    return 0;
}
```

digrafo.txt (57 Bytes)

```
5
0 0 0 0 0
0 0 1 0 0
1 0 1 0 0
1 1 0 0 0
0 1 0 0 0
```

Gravação de Grafo em Arquivo Binário

```
void salvaGrafo(Grafo* g, char* nomeDoArquivo){
    if (!g) return;
    FILE *arq;
    if (!(arq = fopen(nomeDoArquivo, "w"))){
        printf("Nao foi pos...: '%s'\n", nomeDoArquivo);
        exit(1);
    }

    ...
}
```

Gravação de Grafo em Arquivo Binário

```
int x,y,i,cont=0;
int pot[] = {128,64,32,16,8,4,2,1};
int total = (g->numVertices*g->numVertices-1)/8 + 1;
char arranjo[total];
for (i=0;i<total;i++) arranjo[i] = 0;
for (x=0;x<g->numVertices;x++){
    for (y=0;y<g->numVertices;y++){
        arranjo[cont/8] += g->matriz[x][y]*pot[cont%8];
        cont++;
    }
}
fwrite(&(g->numVertices), 1, 1, arq);
for (i=0;i<total;i++)
    fwrite(&(arranjo[i]), 1, 1, arq);
fclose(arq);
}
```

digrafo2.bin (5 Bytes)

00000101 00000001 00101001 10000100 00000000

digrafo2.bin (5 Bytes)

00000101 00000001 00101001 10000100 00000000

5

0 0 0 0 0

0 0 1 0 0

1 0 1 0 0

1 1 0 0 0

0 1 0 0 0

Registros de Tamanho Fixo

```
#define T_nome 30
#define T_email 25

typedef struct{
    int nusp;
    char nome[T_nome];
    int anoDeIngresso;
    char email[T_email];
} PROF;
```

Gravação de Registros de Tamanho Fixo

```
void salvarEmArquivo(char* nomeDoArquivo, PROF* prof){
    FILE *arq;
    if (!(arq = fopen(nomeDoArquivo, "w"))){
        printf("Nao foi pos....: '%s'\n", nomeDoArquivo);
        exit(1);
    }
    fwrite(prof, sizeof(PROF), 1, arq);
    fclose(arq);
}
```

Leitura de Registros de Tamanho Fixo

```
void lerArquivo(char* nomeDoArquivo){
    FILE *arq;  char c;
    if (!(arq = fopen(nomeDoArquivo, "r"))){
        printf("Nao foi ... : '%s'\n", nomeDoArquivo);
        exit(1);
    }
    printf("\nExibindo ... : '%s'\n", nomeDoArquivo);
    char* buffer = (char*)malloc(sizeof(PROF)+1);
    while (!feof(arq)){
        if (fgets(buffer, sizeof(PROF)+1, arq)){
            imprimirProfessor((PROF*)buffer);
        }
    }
    fclose(arq);
}
```

Registros de Tamanho Fixo

```
int main(){
    PROF digi = novoProfessor(3140792, "Luciano Antonio
        Digiampietri", 2008, "digiampietri@usp.br");
    salvarEmArquivo("TamanhoFixo.bin", &digi);
    PROF norton = novoProfessor(12345678, "Norton Trevisan
        Roman", 2010, "norton@usp.br");
    adicionarEmArquivo("TamanhoFixo.bin", &norton);
    PROF daniel = novoProfessor(7654321, "Daniel de
        Angelis Cordeiro", 2015, "daniel.cordeiro@usp.br");
    adicionarEmArquivo("TamanhoFixo.bin", &daniel);
    lerArquivo("TamanhoFixo.bin");
    return 0;
}
```

Registros de Tamanho Fixo

Exibindo conteudo do arquivo: 'TamanhoFixo.bin'

3140792	Luciano Antonio Digiampietri	2008	...
12345678	Norton Trevisan Roman	2010	...
7654321	Daniel de Angelis Cordeiro	2015	...

Registros de Tamanho Variável

```
typedef struct{
    char* nome;
    int nusp;
    int anoDeIngresso;
    char* email;
} PROF;
```

Gravação de Registros de Tam. Variável

```
void salvarEmArquivo(char* nomeDoArquivo, PROF* prof){
    FILE *arq;
    if (!(arq = fopen(nomeDoArquivo, "w"))){
        printf("Nao foi possivel abrir ...: '%s'\n", nomeDoArquivo);
        exit(1);
    }
    int x, temp = prof->nusp; int tamanho = 4 + 1;
    while (temp>9){
        temp/=10;
        tamanho++;
    }
    for (x=0;prof->nome[x] != '\0';x++) tamanho++;
    for (x=0;prof->email[x] != '\0';x++) tamanho++;
    fprintf(arq, "%i|i|%s|i|%s|", (tamanho+5), prof->nusp,
        prof->nome, prof->anoDeIngresso, prof->email);
    fclose(arq);
}
```

Leitura de Registros de Tamanho Variável

```
void lerArquivo(char* nomeDoArquivo){
    FILE *arq;
    char c;
    if (!(arq = fopen(nomeDoArquivo, "r"))){
        printf("Nao foi possivel ...: '%s'\n", nomeDoArquivo);
        exit(1);
    }
    printf("\nExibindo conteudo do arquivo: '%s'\n", nomeDoArquivo);
    PROF atual;
    char* buffer = (char*)malloc(sizeof(char)*256);
    int tamanho, i, x = 0;
    while (!feof(arq)){
        if (!fscanf(arq,"%i|",&tamanho)) break;
        fscanf(arq,"%i|",&(atual.nusp));
        x = 0;
        while (fscanf(arq,"%c",&c) != EOF && c != '|'){
            buffer[x] = c;
            x++;
        }
    }
}
```

Leitura de Registros de Tamanho Variável

```
    atual.nome = (char*)malloc(sizeof(char)*(x+1));
    for (i=0;i<x;i++) atual.nome[i] = buffer[i];
    atual.nome[x] = '\0';
    fscanf(arq,"%i|",&(atual.anoDeIngresso));
    x = 0;
    while (fscanf(arq,"%c",&c) != EOF && c != '|'){
        buffer[x] = c;
        x++;
    }
    atual.email = (char*)malloc(sizeof(char)*(x+1));
    for (i=0;i<x;i++) atual.email[i] = buffer[i];
    atual.email[x] = '\0';
    imprimirProfessor(&atual);
    free(atual.email);
    free(atual.nome);
    if (fscanf(arq,"%c",&c) == EOF) break;
}
fclose(arq); free(buffer);
}
```

Registros de Tamanho Variável

```
int main(){
    PROF digi = novoProfessor(3140792, "Luciano Antonio
        Digiampietri", 2008, "digiampietri@usp.br");
    salvarEmArquivo("TamanhoVariavel.txt", &digi);
    PROF norton = novoProfessor(12345678, "Norton Trevisan Roman",
        2010, "norton@usp.br");
    adicionarEmArquivo("TamanhoVariavel.txt", &norton);
    PROF daniel = novoProfessor(7654321, "Daniel de Angelis
        Cordeiro", 2015, "daniel.cordeiro@usp.br");
    adicionarEmArquivo("TamanhoVariavel.txt", &daniel);
    lerArquivo("TamanhoVariavel.txt");
    return 0;
}
```

Registros de Tamanho Variável

```
63|3140792|Luciano Antonio Digiampietri|2008|digiampie  
tri@usp.br|51|12345678|Norton Trevisan Roman|2010|nort  
on@usp.br|64|7654321|Daniel de Angelis Cordeiro|2015|d  
aniel.cordeiro@usp.br|
```

Algoritmos e Estruturas de Dados II

Aula 17 – Arquivos: Exemplos de Códigos

Prof. Luciano A. Digiampietri
digiampietri@usp.br
@digiampietri

2025