



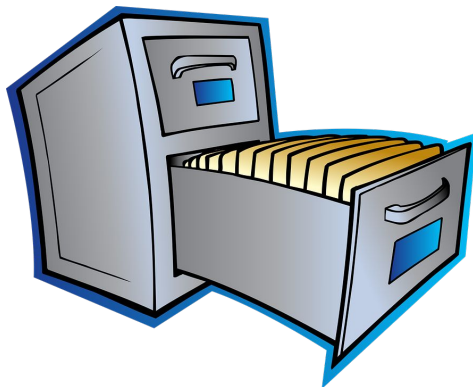
Algoritmos e Programação II

Prof.^a Noeli A. Pimentel Vaz
Prof. Joilson dos Reis Brito

Roteiro da Aula

- ✓ Correção exercícios Slide13
- ✓ Continuação Arquivos
- ✓ Desenvolvimento de Exercícios - Lista08

CONTINUAÇÃO ARQUIVOS



Leitura de strings de Arquivo

fgets

Função que realiza a leitura de uma string em um arquivo.

Protótipo:

char *fgets (char *str, int tamanho, FILE *fp);

A função fgets recebe 3 argumentos:

- a string a ser lida,
- o limite máximo de caracteres a serem lidos;
- e o ponteiro para FILE, que está associado ao arquivo de onde a string será lida.

Leitura de strings de Arquivo

- A função lê a string até que um caracter de nova linha seja lido ou tamanho-1 caracteres tenham sido lidos.
- Se o caracter de nova linha ('\n') for lido, ele fará parte da string, o que não acontecia com gets.
- A string resultante sempre terminará com '\0' (por isto somente tamanho-1 caracteres, no máximo, serão lidos).

Exemplo 1

```
#include<stdio.h>
#include<locale.h>
#include<stdlib.h>
int main()
{
    FILE *PonteiroArquivo;
    char String[100];
    char NomeArquivo[100];
    setlocale(LC_ALL,"Portuguese");
    printf("Digite o nome do arquivo: ");
    gets(NomeArquivo);
    ponteiroArquivo = fopen(NomeArquivo,"r");
    if(PonteiroArquivo == NULL)
        printf("Impossível abrir o arquivo %s!\n",NomeArquivo);
    else
    {
        printf("Palavras armazenadas no arquivo: \n");
        while(!feof(PonteiroArquivo))
        {
            fgets(String, 99, PonteiroArquivo);
            printf("%s",String);
        }
        fclose(PonteiroArquivo);
        printf("\n");
        system("Pause");
    }
    return 0;
}
```

Escrita de strings de Arquivo

fputs

Função que escreve uma string em um arquivo.

Protótipo:

char *fputs (char *str, FILE *fp);

A função grava a string **str** no arquivo **fp**.

Verificação de Erros ao Acessar o Arquivo

ferror

Verifica se um acesso ao arquivo obteve sucesso.

Protótipo:

int ferror (FILE *fp);

A função retorna:

- zero, se nenhum erro ocorreu;
- um número diferente de zero se algum erro ocorreu durante o acesso ao arquivo.

Verificação de Erros ao Acessar o Arquivo

error

Se torna muito útil quando queremos verificar se cada acesso a um arquivo teve sucesso, de modo que consigamos garantir a integridade dos nossos dados. Na maioria dos casos, se um arquivo pode ser aberto, ele pode ser lido ou gravado. Porém, existem situações em que isto não ocorre.

Por exemplo, pode acabar o espaço em disco enquanto gravamos, ou o disco pode estar com problemas e não conseguimos ler, etc.

Exemplo 2

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

#include <locale.h>int main()
{
    FILE *PonteiroArquivo;
    char String[100];
    PonteiroArquivo = fopen("Arquivo.txt","w");
    if(PonteiroArquivo == NULL)
    {
        printf("\n Nao consigo abrir o arquivo ! ");
        exit(1);
    }
    do
    {
        printf("\nDigite uma nova string. Para terminar, digite <enter>: ");
        gets(String);
        if(String[0] != '\n')
        {
            strcat(String,"\n");
            fputs(String, PonteiroArquivo);
        }
        if(ferror(PonteiroArquivo))
        {
            printf("Erro na gravacao\n");
            fclose(PonteiroArquivo);
            exit(1);
        }
    } while (String[0] != '\n');
    fclose(PonteiroArquivo);
    system("Pause"); return 0;
}
```

E se fosse necessário adicionar neste mesmo código a impressão do arquivo que foi lido?

Voltar para o Início do Arquivo

rewind

Retorna a posição corrente do arquivo para o início.

Protótipo:

void rewind (FILE *fp);

Exemplo 3

```
{
FILE *PonteiroArquivo;
char String[100];
PonteiroArquivo = fopen("frutas.txt","a+");
if(PonteiroArquivo == NULL)
{
    printf("\n Nao consigo abrir o arquivo ! ");
    exit(1);
}
do
{
    printf("\nDigite o nome de uma fruta. Para terminar, digite <enter>: ");
    gets(String);
    if(String[0] != '\0')
    {
        strcat(String,"\n");
        fputs(String, PonteiroArquivo);
        if(ferror(PonteiroArquivo))
        {
            printf("Erro na gravacao\n");
            fclose(PonteiroArquivo);
            exit(1);
        }
    }
} while (String[0] != '\0');
system("cls");
printf("Frutas armazenadas no arquivo: \n");
rewind(PonteiroArquivo);
while(!feof(PonteiroArquivo))
{
    fgets(String, 99, PonteiroArquivo);
    if(!feof(PonteiroArquivo))
        printf("%s",String);
}
fclose(PonteiroArquivo);
system("Pause");
return 0;
}
```