UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

Fundamentos de Programação

CP41F

Operações de abertura, leitura e finalização de arquivos.

Aula 14

Prof. Daniel Cavalcanti Jeronymo

Universidade Tecnológica Federal do Paraná (UTFPR)
Engenharia de Computação – 1º Período
2016.1

Plano de Aula

- Arquivos.
- Operações de abertura.
- Finalização de arquivos.
- Operações de leitura.

Arquivos

- Arquivos são sequências de bytes armazenados em disco.

- Nenhuma estrutura é imposta sobre arquivos, isso é deixado para o criador do arquivo.

- Faça o teste:
  - Crie um arquivo texto “teste.txt”
  - Escreva algum conteúdo
  - Abra-o com o CodeBlocks no modo de editor hexadecimal ou outro aplicativo similar

Arquivos

- Utilizando uma visualização de 16 colunas:

Índice	Conteúdo em bytes, representação hexadecimal																Conteúdo em caracteres
00000000:	54	65	73	74	61	6E	64	6F	2C	20	61	71	75	69	20	76	Testando, aqui v
00000010:	61	69	20	75	6D	61	20	71	75	65	62	72	61	20	64	65	ai uma quebra de
00000020:	20	6C	69	6E	68	61	3A	0D	0A	45	20	61	71	75	69	20	linha: E aqui
00000030:	76	61	69	20	75	6D	20	63	61	72	61	63	74	65	72	65	vai um caractere
00000040:	20	74	61	62	3A	09	66	69	6E	61	6C	20	64	6F	20	74	tab: final do t
00000050:	61	62	2C	20	6F	75	74	72	61	20	71	75	65	62	72	61	ab, outra quebra
00000060:	20	64	65	20	6C	69	6E	68	61	3A	0D	0A	66	69	6D	21	de linha: fim!

- Localize os caracteres para quebra de linha. Quantos caracteres e quais são?
- Localize o caractere de tabulação (tab). Qual é?
- Quais são as sequências de escape desses caracteres em C?

Operações de abertura

- Um arquivo deve ser aberto antes que seja possível efetuar operações de leitura ou escrita.

- Quando um arquivo é aberto, um descritor é associado a esse arquivo.

- A abertura com sucesso de um arquivo retorna um ponteiro para uma estrutura de arquivo, que contém o descritor e um bloco de controle.

Operações de abertura

- Para abertura de arquivos, deve-se usar a função:

```
FILE * fopen (const char * filename, const char * mode)
```

- A função retorna um ponteiro para uma estrutura FILE em caso de sucesso ou NULL em caso de erro.
- O usuário não deve se preocupar com a implementação de FILE, o retorno de fopen é um ponteiro opaco (opaque pointer).
- O parâmetro filename especifica o caminho do arquivo.
- O parâmetro mode especifica o modo de abertura do arquivo.

Operações de abertura

- O parâmetro mode pode ser um dos seguintes:

Modo			Descrição	Origem
r	rb		Abre como leitura (arquivo deve existir)	Início
w	wb		Abre como escrita (cria o arquivo se não existir, sobrescreve o arquivo caso exista)	Início
a	ab		Abre como apêndice (cria o arquivo se não existir)	Fim
r+	rb+	r+b	Abre como leitura e escrita (arquivo deve existir)	Início
w+	wb+	w+b	Abre como leitura e escrita (cria o arquivo se não existir, sobrescreve o arquivo caso exista).	Início
a+	ab+	a+b	Abre como leitura e escrita (apêndice caso o arquivo exista, cria o arquivo se não existir)	Fim

- r – leitura; w – escrita; a – apêndice; + - atualização; b – binário

- Por padrão, os arquivos são abertos como texto a não ser que o modo binário seja especificado.

Operações de abertura

- Caso o arquivo seja aberto com sucesso, o valor retornado será um ponteiro para uma estrutura FILE.

- Em caso de erro, o valor retornado será NULL.

- Assumindo que “meusdados.txt” não existe:

```
FILE *fp = fopen(“meusdados.txt”, “r”);
if(fp == NULL)
{
    printf(“O arquivo meusdados.txt não existe!”);
    return 0;
}
```

Finalização de arquivos

- Após ser aberto, o arquivo deve ser finalizado. Modifique o código anterior!

- Isso deve ser realizado pela função fclose:

```
int fclose (FILE * stream)
```

- Disassocia o arquivo especificado por stream.

- Todos os buffers associados são disassociados e “esvaziados”. Operações pendentes de escrita são realizadas e conteúdo nos buffers de entrada é descartado.

Finalização de arquivos

- fclose retorna zero em caso de sucesso e EOF em caso de erro.

- A macro EOF é utilizada para representar o fim de arquivo ou, como nesse caso, para sinalizar a ocorrência de um erro.

- No caso de operações pendentes de escrita, fclose pode falhar caso essas operações não sejam realizadas com sucesso, por exemplo, no evento de erro de entrada/saída ou falta de espaço em disco.

Operações de leitura

- Para ler a partir de arquivos, as funções apropriadas devem ser utilizadas.

- Em modo texto os modos de leitura e entrada são similares aos do console.

- Funções de interesse:
  - fgetc, fgets, fscanf, fread

Operações de leitura

- A ocorrência do final de arquivo pode, e deve, ser verificada pela função feof:

```
int feof(FILE * stream)
```

- A função retorna zero caso o indicador de final de arquivo não esteja definido.

- A função retorna diferente de zero caso o final de arquivo seja alcançado.

Operações de abertura

- Arquivos podem ser abertos em modo texto ou binário.
- Em modo texto, a plataforma é responsável por realizar traduções em operações de leitura e escrita.
- Em modo binário, o conteúdo lido é exatamente o presente no arquivo.

Operações de leitura

- Crie um arquivo texto com o seguinte conteúdo: Teste1<entre com duas novas linhas>Teste2

- Escreva um programa que leia caractere a caractere até alcançar o fim de arquivo, imprimindo para cada caractere seu valor hexadecimal e o caractere em si.

- Teste primeiro com o arquivo em modo de leitura texto “r”, e depois em modo de leitura binária “rb”. Qual a diferença?

Operações de leitura

- Exemplo:

```
#include <stdio.h>

int main()
{
    FILE *fp = fopen("teste.txt", "r");

    while(!feof(fp))
    {
        int c = fgetc(fp);
        printf("Valor: %02X Caractere:
%c\n",c,c);
    }

    fclose(fp);

    return 0;
}
```

Operações de leitura

- Arquivos de apenas leitura não podem ser escritos!

- Tentativas de escrita resultam em erros.

- Modifique o código anterior para que uma tentativa de escrita seja realizada e observe os resultados.

Operações de leitura

- Faça um programa que leia um arquivo com N linhas, cada linha com 3 colunas de inteiros e imprima a média de cada linha.

- Por exemplo, para o arquivo:

  5 9 3

  1 -1 0

- Saída: 5.67 0