**MAC2166 Introdução à Computação para Engenharia**

**Manipulação de arquivos**

Até agora, todos os programas feitos nesta disciplina lêem dados do teclado (*entrada padrão*) e escrevem na tela do monitor (*saída padrão*). Aqui é mostrado como um programa em C lê dados de um arquivo que está no disco (HD ou DISQUETE) e como grava um arquivo no disco. Um tratamento profundo do assunto, discutido brevemente aqui, pode ser encontrado no livro:

The C programming language  
Brian W. Kernigham e Dennis M. Ritchie  
Prentice-Hall

**Abertura de um arquivo (fopen)**

Neste ponto tratamos a seguinte questão:

Como fazer que um arquivo em disco, que tem o seu nome *externo* ao programa, seja associado a algum objeto ou variável que é *interna* ou conhecida do nosso programa?

A regra é clara (como diria o Armando Marques :-) ) Antes que possa ser lido ou gravado um arquivo deve ser *aberto* pela função de biblioteca fopen. A função fopen associa o nome de uma arquivo externo ao programa, como entrada.txt, a uma variável do programa. Isto é feito da seguinte maneira

FILE \*arq\_entrada;

[. . .]

arq\_entrada = fopen("entrada.txt","r");

[. . .]

O primeiro parâmetro da função fopen é uma cadeia de caracteres (*string*, seja lá o que for isto) contendo o nome do arquivo externo. Já o segundo parâmetro, que também é uma cadeia de caracteres, indica o *modo* como se pretende usar o arquivo: leitura ("r"), gravação ("w"), anexação ("a"). Por exemplo o comando

FILE \*arq\_entrada;

[. . .]

arq\_entrada = fopen("entrada.txt","r");

if (arq\_imagem == NULL)

{

printf("Um erro ocorreu ao tentar abrir o arquivo `mamute.pgm'.\n");

}

[. . .]

associa a variável arq\_entrada do nosso programa o arquivo em disco de nome entrada.txt. O arquivo entrada.txt deve estar na mesma pasta (diretório, folder) que o nosso programa, digamos, o ep3.exe. O arquivo é aberto para *leitura* ("r" de *read*). Se ocorrer um erro, como tentar associar arq\_entrada a um arquivo que não existe, fopen retorna o valor representado pela constante NULL.

Se um arquivo que não existe no disco for aberto para escrita, ele é criado. Por exemplo, o seguinte trecho de código cria um arquivo de nome saida.txt:

FILE \*arq\_saida;

[. . .]

arq\_saida = fopen("saida.txt","w");

if (arq\_saida == NULL)

{

printf("Um erro ocorreu ao tentar criar o arquivo `saida.txt'\n");

}

[. . .]

O arquivo saida.txt é criado na mesma pasta/diretório que o arquivo que contém o nosso programa, suponha que seja o ep3.exe. Se o arquivo já existe ele é destruído e um novo arquivo com o mesmo nome é criado. O arquivo é aberto para escrita ("w" de write).

Sempre que um erro ocorre durante a abertura de um arquivo o valor devolvido pela função fopen é NULL.

**Leitura de um arquivo (fscanf)**

O próximo passo é saber como ler os dados de uma arquivo. Considere o arquivo entrada.txt que foi associado à variável arq\_entrada do nosso programa. O comando

FILE \*arq\_entrada;

[. . .]

arq\_entrada = fopen("entrada.txt","r");

[. . .]

fscanf(arq\_entrada,"%d", &num);

[. . .]

lê um número inteiro do arquivo entrada.txt da mesma maneira que

[. . .]

scanf("%d", &num);

[. . .]

lê um número digitado através do teclado. Na verdade, o comando

scanf("%d",&num);

é uma abreviatura do comando

fscanf(stdin,"%d", &num);

onde stdin é a *entrada padrão*, que no nosso caso, é o teclado.

**Escrita em arquivo (fprintf)**

Para escrever-se em um arquivo de nome saida.txt pode-se usar os seguintes comandos

FILE \*arq\_saida;

[. . .]

arq\_saida = fopen("saida.txt","w");

[. . .]

fprintf(arq\_saida,"Copia\n");

fprintf(arq\_saida,"%d %d\n", n, m);

[. . .]

O primeiro fprintf escreve o texto "Copia\n" no arquivo saida.txt e o segundo fprintf escreve o conteúdo da variável n, um espaco (" "), o conteúdo da variável m e um barra-n ("\n") para mudar de linha.

Veja um exemplo um pouco mais elaborado:

FILE \*arq\_saida;

int n;

int m;

[. . .]

arq\_saida = fopen("saida.txt","w");

if (arq\_saida == NULL)

{

printf("Um erro ocorreu ao tentar criar o arquivo `saida.txt'.\n");

}

[. . .]

fprintf(arq\_saida,"Copia\n");

fprintf(arq\_saida,"%d %d\n", n, m);

fprintf(arq\_saida,"n\*m=%d\n", n\*m);

[. . .]

De maneira semelhante ao que ocorre com o scanf, o comando

printf("%d\n", valmax);

é uma abreviatura do comando

fprintf(stdout,"%d\n", valmax);

onde stdout é a *saída padrão*, que no nosso caso, é a tela do monitor.

**Fechamento de um arquivo (fclose)**

A função fclose encerra a associação estabelecida pelo programa entre uma variável e o nome externo do arquivo. O fclose também libera os recursos do sistema que controlam a manipulação do arquivo em disco. A chamada abaixo *fecha* o arquivo associado à variável arq\_saida.

fclose(arq\_saida);

**Exemplo**

A seguir está um exemplo de programa que copia números de um arquivo de nome original.dat para um arquivo de nome copia.dat. O primeiro número do arquivo original.dat indica o número n de números no arquivo.

/\*

\* Programa que que faz uma copia de um arquivo

\* de nome original.dat para um arquivo de nome copia.dat. O

\* arquivo copia.dat ficara no mesmo diretorio do arquivo

\* original.dat.

\*

\*/

#include <stdio.h>

#include <stdlib.h>

int main()

{

FILE \*arq\_o; /\* associado ao arquivo original \*/

FILE \*arq\_c; /\* associado ao arquivo copia \*/

int n; /\* numero de elementos do arquivo original \*/

float x;

int i;

/\* 1. abra arquivo original.dat para leitura \*/

arq\_o = fopen("original.dat","r");

if (arq\_o == NULL)

{

printf("Erro na abertura do arquivo original.dat.\n");

system("pause"); /\* para WINDOWNS \*/

exit(-1); /\* abandona a execucao do programa \*/

}

/\* 2. abra arquivo copia.dat para escrita \*/

arq\_c = fopen("copia.dat","w");

if (arq\_c == NULL)

{

printf("Erro na abertura do arquivo copia.dat.\n");

system("pause"); /\* para WINDOWNS \*/

exit(-1); /\* abandona a execucao do programa \*/

}

/\* 3. leia o tamanho da sequencia no arquivo original.dat \*/

fscanf(arq\_o,"%d", &n);

/\* 4. escreva o numero de elementos no arquivo copia.dat \*/

fprintf(arq\_c,"%d\n", n);

/\* 5. leia os numeros do arquivo original.dat e escreva em copia.dat\*/

for (i = 0; i < n; i++)

{

fscanf(arq\_o,"%f", &x);

fprintf(arq\_c,"%f ", x);

if (i == 5)

{ /\* apenas cinco numeros por linha de arq\_c \*/

fprintf(arq\_c,"\n");

}

}

/\* 6. feche o arquivo original.dat \*/

fclose(arq\_o);

/\* 7. feche o arquivo copia.dat \*/

fclose(arq\_c);

return 0;

}