

MC102 - Algoritmos e Progração de Computador

Prof. Alexandre Xavier Falcão

24° Aula: Arquivos (cont.)

1 Arquivo binário

No caso de arquivos binários, os comandos **fread** e **fwrite** são usados para leitura e escrita de seqüências de bytes. Se os arquivos de leitura e gravação do programa anterior fossem binários, as funções de leitura e gravação seriam conforme abaixo.

```
Agenda LeAgenda(char *nomearq)
{
    int i;
    FILE *aparq;
    Agenda a;

    aparq = fopen(nomearq,"rb"); /* abre arquivo binario para leitura */

    /* le 1 elemento do tamanho de um inteiro copiando sua sequencia de
       bytes para a memoria a partir do endereco apontado por
       &a.num_amigos. Avanca o respectivo numero de bytes na posicao para a
       proxima leitura. */

    fread(&a.num_amigos,sizeof(int),1,aparq);

    a.amigo = (Amigo *)calloc(a.num_amigos,sizeof(Amigo));
    for (i=0; i < a.num_amigos; i++){
        printf("%d\n",ftell(aparq)); /* So uma curiosidade. Diz a posicao no arquivo para leitura */
        fread(&a.amigo[i],sizeof(Amigo),1,aparq);
    }
    fclose(aparq); /* fecha arquivo */
return(a);
}

void GravaAgenda(Agenda a, char *nomearq)
{
    int i;
    FILE *aparq;

    aparq = fopen(nomearq,"wb"); /* cria novo arquivo binario aberto
```

```

para escrita */

/* escreve 1 elemento do tamanho de um inteiro na posicao atual de
   escrita no arquivo, copiando sua sequencia de bytes da memoria a
   partir do endereco apontado por &a.num_amigos. Avanca o respectivo
   numero de bytes na posicao para a proxima escrita. */

fwrite(&a.num_amigos,sizeof(int),1,aparq);

for (i=0; i < a.num_amigos; i++){
    fwrite(&a.amigo[i],sizeof(Amigo),1,aparq);
}
fclose(aparq); /* fecha arquivo */
}

```

Arquivos biários também podem ser modificados sem necessariamente terem que ser lidos e escritos por completo no disco. As funções abaixo ilustram uma alteração no número de telefone do segundo registro e a adição de um novo registro no final do arquivo.

```

void AlteraRegistro(char *nomearq)
{
    FILE *aparq;
    Amigo aux;

    /* abre arquivo existente para leitura/escrita */
    aparq = fopen(nomearq,"r+");

    /* posiciona leitura/escrita no inicio do segundo registro */
    fseek(aparq,sizeof(int)+sizeof(Amigo),SEEK_SET);

    fread(&aux,sizeof(Amigo),1,aparq); /* le nome e telefone */
    aux.telefone = 10; /* Altera telefone para o novo numero */

    /* posiciona leitura/escrita no inicio do segundo registro */
    fseek(aparq,sizeof(int)+sizeof(Amigo),SEEK_SET);

    fwrite(&aux,sizeof(Amigo),1,aparq); /* grava registro modificado */
    fclose(aparq); /* fecha arquivo */
}

void AdicionaRegistro(char *nomearq)
{
    FILE *aparq;
    Amigo aux;
    int num_amigos;

```

```

/* abre arquivo existente para leitura/escrita. Posiciona escrita no
   final do arquivo, mas permite leitura dos registros se o apontador for
   reposicionado. So permite escrita no final.*/
aparq = fopen(nomearq,"a+");

sprintf(aux.nome,"Gloria Meneses");
aux.telefone = 65;
fwrite(&aux,sizeof(Amigo),1,aparq); /* Grava no final */
fseek(aparq,0,SEEK_SET); /* posiciona leitura no inicio do arquivo */
fread(&num_amigos,sizeof(int),1,aparq); /* le numero de amigos */
num_amigos ++; /* atualiza o numero de amigos */
fclose(aparq); /* fecha arquivo */
/* abre para leitura/escrita em qualquer posicao */
aparq = fopen(nomearq,"r+");

/* grava numero de amigos atualizado */
fwrite(&num_amigos,sizeof(int),1,aparq);
fclose(aparq); /* fecha arquivo */
}

```

As funções acima gravam o arquivo binário com registros de tamanho fixo. Para o exemplo dado, o arquivo binário ocupa mais espaço do que o arquivo texto. Isto porque estamos gravando 50 bytes do nome de cada amigo, quando este nome ocupa menos que 50 bytes. Uma alternativa é gravar 1 byte com a informação do comprimento do nome e depois o nome. Isto reduz bastante o tamanho do arquivo binário.

```

Agenda LeAgenda(char *nomearq)
{
    Agenda a;
    FILE *aparq;
    int i;
    unsigned char comp;

    aparq = fopen(nomearq,"rb");
    fread(&a.num_amigos,sizeof(int),1,aparq);
    a.amigo = (Amigo *) calloc(a.num_amigos,sizeof(Amigo));
    if (a.amigo != NULL){
        for (i=0; i < a.num_amigos; i++) {
            fread(&comp,sizeof(unsigned char),1,aparq);
            fread(a.amigo[i].nome,comp,1,aparq);
            fread(&a.amigo[i].telefone,sizeof(int),1,aparq);
        }
    }
    fclose(aparq);
    return(a);
}

```

```
void GravaAgenda(Agenda a, char *nomearq)
{
    int i;
    unsigned char comp;
    FILE *aparq;

    aparq = fopen(nomearq,"wb");
    fwrite(&a.num_amigos,sizeof(int),1,aparq);
    for (i=0; i < a.num_amigos; i++){
        comp = (unsigned char)strlen(a.amigo[i].nome);
        fwrite(&comp,sizeof(unsigned char),1,aparq);
        fwrite(a.amigo[i].nome,comp,1,aparq);
        fwrite(&a.amigo[i].telefone,sizeof(int),1,aparq);
    }
    fclose(aparq);
}
```