

Linguagem C arquivos binários

André Tavares da Silva

andre.silva@udesc.br

Arquivos Binários

- Vimos que existem dois tipos de arquivos: textos e binários.
- Na representação textual os valores precisam ser armazenados na forma "humana" enquanto na representação binária os valores são armazenados conforme representação binária (int, double,...).
- A representação textual os dados normalmente são com tamanhos variados, enquanto na representação binária os dados são preferencialmente armazenados em forma de registros.
- Armazenar dados em arquivos de forma análoga à utilizada em memória nos permite reduzir o tamanho do arquivo e acessar facilmente estruturas complexas.

Arquivos Binários

- Um arquivo binário em C funciona como um vetor com um indicador de posição que revela a qualquer momento, em que ponto o arquivo será acessado. O deslocamento da posição é sempre especificada em bytes e toma como referência o início do arquivo, ou seja, a posição zero.
- Se um arquivo é recém criado seu comprimento é zero e o indicador de posição assim indicará.
- As funções de leitura ou escrita utilizam o indicador de posição para executar suas ações. Cada operação atualiza esse indicador de posição em função do número de bytes lidos ou escritos.

Funções de leitura e escrita em arquivos binários

- As funções **fread** e **fwrite** são utilizadas para ler e escrever blocos de dados de qualquer tipo.
- Através delas é possível escrever ou ler de um arquivo, por exemplo, toda uma matriz, estrutura ou qualquer outro tipo de dado.
- Devemos determinar o número de elementos a serem lidos ou gravados e o tamanho de cada um.

fread e fwrite

```
size_t fwrite(void *ptr, size_t size, size_t num-items, FILE *arq);
```

```
size_t fread(void *ptr, size_t size, size_t num-items, FILE *arq);
```

ptr: ponteiro para região da memória (já alocada) para onde os dados serão copiados (fread) ou contendo os itens a serem gravados no arquivo (fwrite).

size: número de bytes de um item.

num-items: número de itens a serem lidos/gravados.

arq: ponteiro para o arquivo.

fread e fwrite

- Lembre-se do indicador de posição de um arquivo, que assim que é aberto é apontado para o início do arquivo.
- Quando lemos uma determinada quantidade de itens, o indicador de posição automaticamente avança para o próximo item não lido.
- Quando escrevemos algum item, o indicador de posição automaticamente avança para a posição seguinte ao item escrito.

Verificando o fim de arquivo

- Ao ficar lendo os dados de um arquivo precisamos sempre verificar se o final do arquivo foi atingido (não existem mais dados a serem lidos)
- Para isso existe o comando **feof** que retorna não-zero se o arquivo chegou ao final e não-zero caso contrário.

```
int feof(FILE *arq);
```

arq: ponteiro para o arquivo a ser verificado.

Acesso não sequencial

- O acesso não sequencial a arquivos é feito usando a função **fseek**.
- Esta função altera a posição de leitura/escrita no arquivo.
- O deslocamento pode ser relativo a:
 - início do arquivo (**SEEK_SET**)
 - ponto atual (**SEEK_CUR**)
 - final do arquivo (**SEEK_END**)

Acesso não sequencial

```
int fseek(FILE *arq, long num-bytes, int origem);
```

arq: ponteiro para arquivo.

num-bytes: quantidade de bytes para se deslocar.

origem: posição de início do deslocamento
(SEEK_SET, SEEK_CUR ou SEEK_END).

- A função **rewind** retorna a posição corrente do arquivo para o início. O mesmo que **fseek(arq, 0, SEEK_SET)**.

```
void rewind (FILE *arq);
```

arq: ponteiro para o arquivo a ser manipulado.

Acesso não sequencial

- Um arquivo pode armazenar registros (como um banco de dados).
- Isso pode ser feito de forma bem fácil pois qualquer variável em C, tem um tamanho fixo.
- O acesso a cada registro pode ser direto, usando a função **fseek**.
- A leitura ou escrita do registro pode ser feita usando as funções **fread** e **fwrite**.

Exercícios

- Criar um programa em C que faça o gerenciamento de arquivos de tamanho fixo, desenvolvendo inclusão, listagem e alteração de registros. Crie uma estrutura de registro com os seguintes campos: nome, e-mail e telefone.
- Altere o programa anterior para também permitir a exclusão de registros.