

Algoritmos e Estruturas de Dados I

Prof. Lúcio Mauro Pereira

02/10/2024

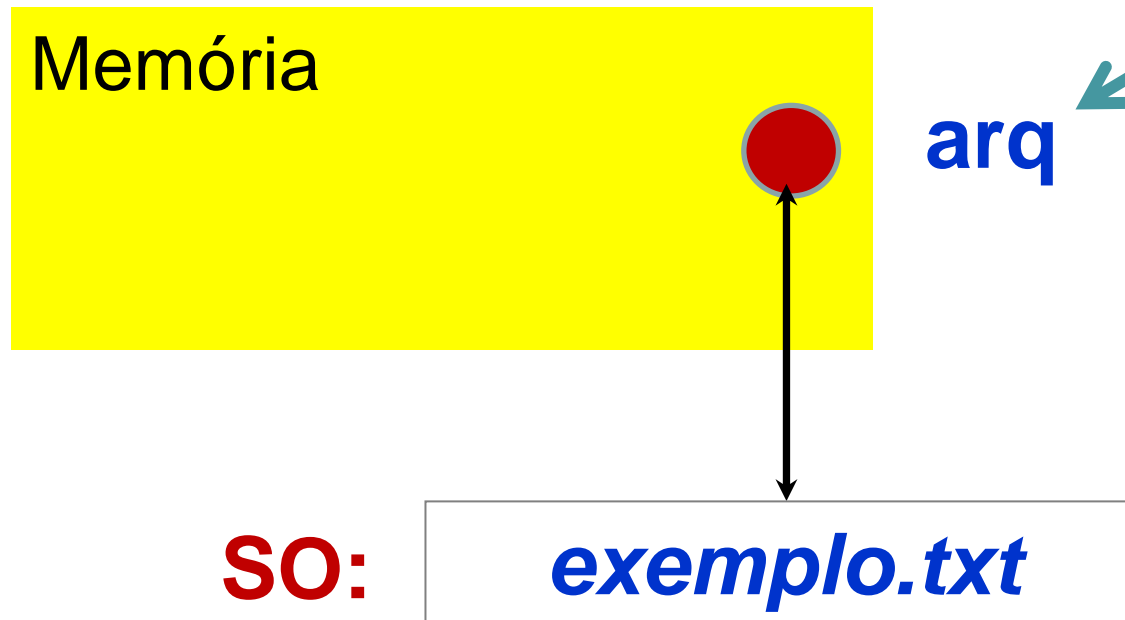
Aula anterior



Criando um novo arquivo

```
FILE* arq = fopen(“exemplo.txt”, “w”);
```

ponteiro para FILE





Modos de abertura de um arquivo com **fopen()**:

w Cria um fluxo para escrita no arquivo - *sobrescreve*

a Cria um fluxo para escrita no arquivo – *ao final*

r Cria um fluxo para leitura do arquivo – *NULL se não houver*

w+ Acrescenta permissão para leitura

a+ Acrescenta permissão para leitura

r+ Acrescenta permissão para escrita



Abrindo, escrevendo e fechando o arquivo

```
FILE *arq;  
arq = fopen( "exemplo.dat", "w");  
  
for(int x=10; x<=20; x+=2){  
    fprintf(arq, "%i\n", x);  
}  
  
fclose(arq);
```


Lendo um arquivo
Fluxo de entrada



fscanf() – Acrescenta o ponteiro para FILE

```
int x;
```

```
scanf( "%i", &x);
```

```
fscanf(stdin, "%i", &x);
```

```
FILE *arq;
```

```
arq = fopen( "exemplo.dat", "r");
```

```
if( arq != NULL ) fscanf(arq, "%i", &x);
```



fscanf() – Acrescenta o ponteiro para FILE

```
FILE *arq;  
if( (arq = fopen( "exemplo.dat", "r")) != NULL )  
{  
    int x;  
    for(int i=1; i<=6; i++){  
        fscanf(arq, "%i", &x);  
        printf("%i\n", x);  
    } // fim for(i)  
    fclose(arq);  
} // fim if
```




feof() – Identifica *end of file* – Função booleana

```
FILE *arq;  
if( (arq = fopen( "exemplo.dat", "r")) != NULL )  
{  
    int x;  
    fscanf(arq, "%i", &x);  
    while( !feof(arq) ){  
        printf("%i\n", x);  
        fscanf(arq, "%i", &x);  
    }  
    fclose(arq);  
}
```



rewind() – Reposiciona no início do arquivo

```
FILE *arq;  
if( (arq = fopen( "exemplo.dat", "r" ) ) != NULL )  
{  
    int x;  
    fscanf(arq, "%i", &x);  
    while( !feof(arq) ){  
        printf("%i\n", x);  
        fscanf(arq, "%i", &x);  
    }  
}  
rewind(arq);  
...
```



rewind() – Reposiciona no início do arquivo

```
int x;
FILE *arq;
if( (arq = fopen( "exemplo.dat", "r" ) ) != NULL ) {
    fscanf(arq, "%i", &x);
    while( !feof(arq) ){
        printf("%i\n", x);
        fscanf(arq, "%i", &x);
    }
    rewind(arq);
    fscanf(arq, "%i", &x);
    while( !feof(arq) ){
        printf("%i\n", x);
        fscanf(arq, "%i", &x);
    }
    fclose(arq);
}
```

Arquivo texto
versus
Arquivo binário



Arquivo texto:

Padrão ASCII (*humano*)

Tipicamente, extensão *.txt*

Arquivo binário:

Mantém representação digital (*máquina*)

Tipicamente, diversas extensões, como *.dat*, *.bin*



Modos de abertura de um arquivo com **fopen()**:

wb Cria um fluxo para escrita no arquivo - *sobrescreve*

ab Cria um fluxo para escrita no arquivo – *ao final*

rb Cria um fluxo para leitura do arquivo – *NULL se não houver*

wb+ ou **w+b** Acrescenta permissão para leitura

ab+ ou **a+b** Acrescenta permissão para leitura

rb+ ou **r+b** Acrescenta permissão para escrita



Tente:

Construa uma função que receba uma *string* relativa ao nome de um arquivo que armazena um conjunto de valores reais.

A função deverá retornar o número de elementos presentes no arquivo cujo valor é maior que a média dos valores nele armazenados.

Modularização: Divida a tarefa em funções com funcionalidades atômicas