

Objetivos	a. Demonstrar o funcionamento de diversas ED sob os aspectos: ocupação de memória, armazenamento e recuperação de conteúdo, operações principais b. Experimentar a leitura e gravação de arquivos em formato texto e binário
Referências gerais	SCHILDT, H. <i>C Completo e Total</i> . São Paulo: Pearson, 1997. TENENBAUM, A.M.; LANGSAM, Y.; AUGENSTEIN, M.J. <i>Estruturas de Dados usando C</i> . São Paulo: Pearson Makron Books, 1995.

Objetivo específico desta aula prática: trabalhar com tipos primitivos, “structs”, arquivos-texto e arquivos binários (linguagem C).

Suporte teórico 1. Funções e operações da linguagem C que permitem saber a quantidade de memória ocupada e o endereço real de cada variável.

Função / operação	Descrição	Exemplo	Obs.
sizeof()	Retorna a quantidade (em bytes) que a variável ou ED ocupa na memória	sizeof(int) sizeof (x)	A quantidade (em bytes) sempre é um número inteiro.
&	Retorna o endereço do seu operando	printf(“%p\n”, &x);	O formato de impressão %p permite a exibição de valores hexadecimais.

Atividade 1. Crie um programa em linguagem C no qual você declara variáveis de todos os tipos primitivos, bem como uma “struct”, atribui valores a elas e imprime, para cada uma, as seguintes informações: o conteúdo atual, a quantidade de bytes ocupados na memória e o respectivo endereço.

Atividade 2. Com base nos resultados impressos na Atividade 1, elabore um desenho que represente a ocupação da memória pelas variáveis do seu programa. A memória deve ser representada como no modelo ao lado.

#endereço			
			← 4 bytes →

Suporte teórico 2. Funções para leitura e gravação em arquivos-texto.

O que	Descrição	Exemplo	Obs.
FILE *	Tipo da variável que será associada ao arquivo em disco	FILE * arquivo;	-
fopen() fclose()	Funções para “abrir” o arquivo (torná-lo disponível para o programa) e depois fechá-lo	arquivo = fopen(“arq.txt”, “r”); fclose(arquivo);	Pode ser: “r”, “w”, “a”
fscanf()	Lê do arquivo, como se fosse leitura do teclado	fscanf(arquivo, “%d”, &n1);	-
fprintf()	Grava no arquivo, como se fosse impressão em tela	fprintf(arquivo, “%d”, n2);	-

Atividade 3. Diretamente pelo sistema operacional, crie um arquivo-texto vazio. Por meio de um programa em linguagem C, abra esse arquivo, grave conteúdos de tipos primitivos variados e feche-o. Verifique o conteúdo gravado por meio do “bloco de notas” ou similar.

Atividade 4. Com o “bloco de notas” ou similar, altere os conteúdos gravados no arquivo. Escreva um programa em linguagem C que abre o arquivo, lê seu conteúdo, fecha-o e mostra na tela os valores lidos.

Suporte teórico 3. Funções para leitura e gravação em arquivos que não são (unicamente) texto.

O que	Descrição	Exemplo	Obs.
fread()	Lê blocos de tamanho definido (“structs”, por exemplo) no arquivo	fread(&x, sizeof(x), qtd, arq);	-
fwrite()	Grava blocos de tamanho definido (“structs”, por exemplo) no arquivo	fwrite(&x, sizeof(x), qtd, arq);	-

Atividade 5. Abra um arquivo no modo “gravação”, grave uma “struct” e feche esse arquivo.

Atividade 6. Abra o arquivo novamente no modo “leitura” e leia seu conteúdo, trazendo as variáveis para o programa. Exiba o conteúdo dessas variáveis na tela.