INSERÇÃO EM VETOR NÃO ORDENADO

#include <stdio.h>

#define TAMANHO 10

int main()

{

int vetor[TAMANHO] = {10, 30, 20};

int n = 0;

vetor[n++] = 10;

vetor[n++] = 30;

vetor[n++] = 20;

printf("Elementos do vetor:\n");

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

printf("Posição %d: %d\n", i, vetor[i]);

}

return 0;

}

PESQUISA LINEAR EM VETOR NÃO ORDENADO

#include <stdio.h>

#define TAMANHO 10

int main()

{

int vetor[TAMANHO] = {10, 30, 20};

int n = 3;

int valor = 30;

int encontrado = -1;

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

if (vetor[i] == valor){

encontrado = i;

break;

}

}

if (encontrado != -1) {

printf("Valor %d encontrado na posição %d. \n", valor, encontrado);

}else {

printf("Valor %d não encontrado. \n", valor);

}

return 0;

}

EXCLUSÃO DE UM ELEMENTO NO VETOR NÃO ORDENADO

#include <stdio.h> // Biblioteca padrão de entrada e saída

#define TAMANHO 10 // Define o tamanho máximo do vetor

int main() {

int vetor[TAMANHO] = {10, 30, 20}; // Vetor inicializado com 3 valores

int n = 3; // Número de elementos no vetor

int valor = 30; // Valor que queremos excluir

int posicao = -1; // Variável para armazenar a posição do valor a excluir (inicialmente -1 = não encontrado)

// Pesquisa para encontrar a posição do valor

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

if (vetor[i] == valor) { // Se o valor encontrado for igual ao procurado

posicao = i; // Armazena a posição

break; // Para o laço

}

}

// Verifica se encontrou o valor para excluir

if (posicao != -1) {

// Desloca os elementos após a posição encontrada uma casa para a esquerda

for (int i = posicao; i < n - 1; i++) {

vetor[i] = vetor[i + 1]; // Move o próximo valor para a posição atual

}

n--; // Decrementa o número de elementos no vetor

printf("Valor %d excluído.\n", valor); // Informa que a exclusão foi feita

} else {

printf("Valor %d não encontrado.\n", valor); // Informa que o valor não foi encontrado

}

// Exibe o vetor atualizado

printf("Elementos do vetor após exclusão:\n");

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

printf("Posição %d: %d\n", i, vetor[i]);

}

return 0; // Finaliza o programa

}