

Engenharia de computação - IFPB  
Estrutura de Dados e Algoritmos  
Aluno: Rafael Victor Cordeiro Muniz

```
#include <stdio.h>
#include <stdlib.h>

struct no{
    int dado;
    struct no *esq, *dir;
};

struct no *raiz = NULL;
struct no *inserir(struct no *arvore, int dado);
struct no *enderecoNodo(struct no *arvore, int dado);
struct no *apagar(struct no *arvore);

void main() {
    // Elementos a serem adicionados na arvore
    int Arvore[] = {1,2,3,4,5,6,7,8,9,10};

    // Insercao ajustada pelo tamanho da arvore
    int size = sizeof(Arvore)/sizeof(Arvore[0]);

    for (int i = 0; i < size; i++){
        raiz = inserir(raiz, Arvore[i]);
    }

    int EnderecoBuscado;
    printf("\nDado a ser buscado o endereco: ");
    scanf("%d", &EnderecoBuscado);

    if (enderecoNodo(raiz, EnderecoBuscado)){
        printf("O endereco do numero %d eh %p\n", EnderecoBuscado,
enderecoNodo(raiz, EnderecoBuscado));
    } else {
        printf("O endereco do numero %d eh NULL\n", EnderecoBuscado);
    }
    raiz = apagar(raiz);
}

struct no *inserir(struct no *arvore, int dado){
    if (!arvore) {
        arvore = (struct no*) malloc(sizeof(struct no));
        arvore->dado = dado;
        arvore->esq = arvore->dir = NULL;
    } else if (dado < arvore->dado)
        arvore->esq = inserir(arvore->esq,dado);
    else
        arvore->dir = inserir(arvore->dir,dado);

    return arvore;
}

struct no *enderecoNodo(struct no *arvore, int dado){
    if (arvore) {
        if (arvore->dado == dado)
            return arvore;
        else if (dado < arvore->dado)
```

```
        return enderecoNodo(arvore->esq, dado);
    else
        return enderecoNodo(arvore->dir, dado);
    }
    return NULL;
}

struct no *apagar(struct no *arvore){
    if (arvore) {
        arvore->esq = apagar(arvore->esq);
        arvore->dir = apagar(arvore->dir);
        free(arvore);
    }
    return NULL;
}
```