

Identificação: Pedro Mesquita Maia – 2312664

Descrição: A ideia desse código foi implementar uma árvore binária de busca, onde o foco era inserir e organizar nós usando uma lista de identificadores inteiros. Depois de construir a árvore, o próximo passo era calcular a altura de cada nó e fazer uma travessia pra mostrar os valores junto com as informações de nível e altura de cada nó. Isso serve pra verificar se a árvore foi montada direitinho.

Estrutura: Eu criei quatro funções principais, além da função main: createNode, insert, createBT, calculateHeight e showBT. Basicamente, eu defini um array de identificadores e depois chamei a createBT pra construir a árvore binária com esses valores. Depois, calculei a altura dos nós com calculateHeight e usei showBT pra mostrar os nós na ordem.

Solução: Usei um array fixo com os valores {10, 5, 15, 3, 7, 13, 20, 1, 4, 6} pra preencher a árvore. As funções que eu criei foram:

createNode: Aloca dinamicamente um novo nó, inicializa o valor do nó com o identificador passado e os ponteiros (esquerda, direita e pai) com NULL. Além disso, guarda o nível do nó na árvore.

insert: Essa função cuida de inserir os identificadores na árvore de forma recursiva. Basicamente, ela compara o valor atual com o nó pai e decide se deve ir para o filho esquerdo ou direito. Se o nó for NULL, é ali que o novo nó é inserido. Vai sempre guardando o nível dos nós pra facilitar depois.

createBT: Recebe um ponteiro pra árvore e o array de identificadores. Ela vai chamando a função insert pra cada valor do array, montando a árvore.

calculateHeight: Essa função percorre a árvore recursivamente e calcula a altura de cada nó. Pra isso, ela compara a altura dos filhos esquerdo e direito e guarda a maior.

showBT: Printa a árvore, mostrando o valor do nó, o nível, a altura e o valor do nó pai (se tiver). Ela percorre a árvore em ordem (esquerda, nó atual e direita).

Funcionamento:

Inicializei o array com os valores {10, 5, 15, 3, 7, 13, 20, 1, 4, 6} e passei pra função createBT. Essa função inseriu os nós de acordo com as regras de uma árvore binária de busca. O nó com valor 10 virou a raiz e os outros foram sendo inseridos à esquerda ou direita conforme o valor.

Depois de montar a árvore, a função calculateHeight foi chamada pra calcular a altura de cada nó. A altura é simplesmente o número de arestas do nó até a folha mais distante.

Pra finalizar, a função showBT percorreu a árvore e imprimiu os nós, mostrando o valor do nó, seu nível, sua altura e o identificador do nó pai (ou -1 se for a raiz).

Observações: Um dos maiores desafios foi acertar a recursão, principalmente pra calcular a altura dos nós. Tive que ajustar o código algumas vezes até pegar a lógica certinho. Usei vídeos, fóruns e até o chatGPT pra entender melhor como funcionam essas operações em árvores binárias, porque ainda não é algo tão fluído pra mim.

Compilação e Testes: Pra compilar o código e verificar se tinha algum warning, usei o comando: gcc -Wall -o main main.c. Fiz alguns testes com a lista de identificadores e também com listas menores e maiores. Tudo funcionou como esperado, sem erros de execução e com a árvore sendo montada e exibida corretamente.