

Python Estrutura de Dados

Módulo 07
Aula 08

Estrutura de Dados

Árvores Binárias

Árvores binárias são uma das estruturas de dados mais fundamentais e versáteis em ciência da computação, usadas para modelar relações hierárquicas entre elementos com uma eficiência notável tanto em termos de processamento quanto de memória. Aqui está uma breve revisão de suas principais características e propriedades:

ESTRUTURA

- **Nós e Arestas:** Uma árvore binária é composta por nós conectados por arestas. Cada nó representa um valor ou dado, e as arestas definem a relação pai-filho entre os nós.
- **Nó Raiz:** O nó no topo da árvore, a partir do qual todos os outros nós são descendentes.
- **Subárvores:** Cada nó na árvore pode ser visto como a raiz de sua própria subárvore, composta por seu filho esquerdo, seu filho direito, e todos os descendentes desses filhos.

PROPRIEDADES

Binária: Cada nó tem no máximo dois filhos, conhecidos como filho esquerdo e filho direito.

Profundidade e Altura: A profundidade de um nó é o número de arestas do nó raiz até o nó em questão. A altura de uma árvore é a profundidade máxima entre todos os nós.

Níveis: O nível de um nó é definido pela sua profundidade mais um. O nó raiz está no nível 1.

TIPOS ESPECIAIS

- **Árvore Binária de Busca (BST):** Uma árvore binária onde cada nó segue a propriedade de que todos os valores na subárvore esquerda são menores que o valor do nó, e todos os valores na subárvore direita são maiores.
- **Árvore Binária Completa:** Uma árvore binária onde todos os níveis, exceto possivelmente o último, estão completamente preenchidos, e todos os nós estão o mais à esquerda possível.
- **Árvore Binária Balanceada:** Uma árvore onde a diferença de altura entre as subárvores esquerda e direita de qualquer nó é no máximo 1.

OPERAÇÕES BÁSICAS

- **Inserção:** Adicionar um novo nó na árvore, seguindo as propriedades específicas do tipo de árvore binária.
- **Busca:** Localizar um valor dentro da árvore, percorrendo a estrutura a partir do nó raiz.
- **Deleção:** Remover um nó da árvore, o que pode exigir reorganização para manter as propriedades da árvore.
- **Travessia:** Visitar todos os nós da árvore em uma ordem específica. Existem várias estratégias, incluindo pré-ordem, em ordem, e pós-ordem.

Árvores binárias são amplamente utilizadas em computação devido à sua eficiência e capacidade de representar estruturas de dados complexas, desde simples coleções de dados até algoritmos avançados de busca e ordenação.

Estrutura de Dados

Grafos

Grafos são estruturas de dados que modelam um conjunto de objetos (chamados de vértices ou nós) junto com as relações ou conexões entre esses objetos (chamadas de arestas). Grafos oferecem uma maneira poderosa e flexível de representar diversas situações do mundo real, como redes sociais, sistemas de mapas, organizações de sites, e muito mais, onde elementos estão interconectados de maneiras complexas.

GRAFOS DIRIGIDOS (DIGRAFOS)

- **Definição:** Em grafos dirigidos, as arestas têm uma direção definida, indicando uma relação de mão única entre dois vértices. Isso significa que, se existe uma aresta direcionada do vértice A para o vértice B, então A está conectado a B, mas B não está necessariamente conectado a A. As arestas dirigidas são frequentemente representadas por setas apontando do vértice de origem para o vértice de destino.
- **Aplicações:** Grafos dirigidos são usados em situações onde a direcionalidade das relações entre os elementos é importante. Exemplos incluem a modelagem de ruas de mão única em mapas de trânsito, representação de pré-requisitos em um currículo acadêmico, ou fluxos em redes de computadores.

GRAFOS NÃO DIRIGIDOS

- **Definição:** Grafos não dirigidos consistem em vértices conectados por arestas onde a relação é bidirecional, ou seja, se o vértice A está conectado ao vértice B, então B também está conectado a A. As arestas em grafos não dirigidos são tipicamente representadas por linhas simples que ligam dois vértices.
- **Aplicações:** Esses grafos são adequados para modelar relações simétricas entre objetos. Eles são amplamente utilizados para representar redes sociais (onde uma conexão ou amizade é mútua), redes de comunicação não direcionais, e qualquer cenário onde as conexões não têm uma orientação específica.

DISTINÇÃO FUNDAMENTAL

A principal distinção entre grafos dirigidos e não dirigidos reside na natureza das conexões entre os vértices:

- **Dirigidos:** As conexões têm uma direção específica, representando relações assimétricas. Isso permite a modelagem de cenários onde a relação entre dois elementos não é recíproca.
- **Não Dirigidos:** As conexões são bidirecionais e simétricas, ideal para situações onde a presença de uma relação implica automaticamente uma relação recíproca.

Esta distinção é crucial ao escolher o tipo de grafo para modelar problemas específicos, pois influencia tanto a representação dos dados quanto os algoritmos usados para processar e analisar o grafo.