

B-Ağaçları ve İndeksleme Nedir?

Çalışma Kağıdı

B-Ağaçları, disk tabanlı veri yapıları için optimize edilmiş, birden fazla anahtarı ve alt ağacı olan bir ağaç yapısıdır. İndeksleme ise, veritabanı tablolarındaki verilere daha hızlı erişim sağlamak için kullanılan bir tekniktir ve B-Ağaçları bu indekslerin oluşturulmasında yaygın olarak kullanılır.

Sorular

1. B-Ağaçları hangi tür veri yapıları için optimize edilmiştir?

- A) Bellek içi veri yapıları
- B) Disk tabanlı veri yapıları
- C) Dağıtık veri yapıları
- D) Sıralı veri yapıları

2. Bir B-Ağacı'nın 'kendini dengeleme' özelliği ne anlama gelir?

- A) Ağacın her zaman tam dolu olmasını sağlar.
- B) Ekleme ve silme işlemleri sırasında ağacın yüksekliğini sabit tutar.
- C) Kök düğümün her zaman en büyük anahtara sahip olmasını sağlar.
- D) Tüm yaprakların aynı seviyede olmasını garanti eder.

3. Aşağıdakilerden hangisi B-Ağacı'nın bir özelliği DEĞİLDİR?

- A) Her düğüm birden fazla anahtar içerebilir.
- B) Ağaç her zaman ikili (binary) bir yapıdadır.
- C) Tüm yapraklar aynı seviyededir.
- D) Ağacın yüksekliği logaritmiktir.

4. Bir B-Ağacı'nın temel yapısını ve özelliklerini açıklayınız.

5. B-Ağaçları veritabanı indekslemesinde nasıl kullanılır?

6. B-Ağacı'nın ikili arama ağaçlarından (BST) farkı nedir?

7. Tanımla: B-Ağacı Nedir?

8. Tanımla: B-Ağacı'nın Temel Amacı Nedir?

9. Tanımla: B-Ağacı'nda Düğüm Yapısı Nasıldır?

Cevap Anahtarı

1. B) Disk tabanlı veri yapıları — B-Ağaçları, disk okuma/yazma maliyetinin yüksek olduğu disk tabanlı veri yapıları için optimize edilmiştir.
2. D) Tüm yaprakların aynı seviyede olmasını garanti eder. — Kendini dengeleme özelliği, ekleme ve silme işlemleri sonucunda ağacın yapısının bozulmamasını ve tüm yaprakların aynı derinlikte kalmasını sağlar.
3. B) Ağaç her zaman ikili (binary) bir yapıdadır. — B-Ağaçları ikili değildir; her düğüm birden fazla çocuğa sahip olabilir. İkili arama ağaçları ikilidir.
4. Bir B-Ağacı, her düğümün belirli sayıda anahtar tutabildiği ve bu anahtarların sayısına bağlı olarak belirli sayıda alt ağaca sahip olabildiği bir veri yapısıdır. Ağaç, kendini dengeleyerek her zaman en kısa yoldan yapraklara ulaşılmasını sağlar. Kök düğüm hariç tüm düğümler en az belirli sayıda anahtar bulundurur ve tüm yapraklar aynı seviyededir.
5. Veritabanlarında, bir tablonun belirli sütunları (örneğin, birincil anahtar) üzerinde bir B-Ağacı indeksi oluşturulabilir. Bu ağaç, sütundaki değerleri sıralı bir şekilde tutar. Bir sorgu geldiğinde, veritabanı bu B-Ağacı'nı kullanarak ilgili satırların diskteki konumunu hızla bulur, böylece tüm tabloyu tarama ihtiyacı ortadan kalkar.
6. İkili arama ağaçları her düğümde yalnızca bir anahtar tutarken, B-Ağaçları her düğümde birden fazla anahtar tutabilir. Bu, B-Ağaçlarının daha geniş ve daha sığ olmasını sağlar. Disk G/Ç işlemlerinin maliyetli olduğu durumlarda, daha az disk okuması gerektiren B-Ağaçları daha verimlidir.
7. Diske erişimi optimize eden, çok yollu, kendini dengeleyen bir ağaç veri yapısıdır.
8. Büyük veri kümelerinde disk G/Ç işlemlerini azaltarak veri erişimini hızlandırmaktır.
9. Her düğüm birden fazla anahtar ve bu anahtarlara karşılık gelen alt ağaç işaretçileri içerir.

Bounlu

Tüm kartlar, adım adım çözümler ve AI hoca desteği Notek uygulamasında.
Sınav tarihlerini Promy otomatik hatırlatıcıya çevirir.