

Hash Tabloları ve Karma Fonksiyonları Nedir?

Çalışma Kağıdı

Hash tabloları, karma fonksiyonları aracılığıyla anahtarları dizinlere dönüştürerek veriye hızlı erişim sağlayan bir veri yapısıdır.

Sorular

- Aşağıdakilerden hangisi karma fonksiyonunun temel görevidir?
A) Veriyi şifrelemek
B) Anahtarları benzersiz dizinlere dönüştürmek
C) Veri tabanına kaydetmek
D) Veriyi sıkıştırmak
- Hash tablolarındaki çakışmaları çözmek için kullanılan yöntemlerden biri değildir?
A) Zincirleme
B) Açık Adresleme
C) Veriyi Silmek
D) Doğrusal Sondalama
- Bir hash tablosunda veriye erişim genellikle hangi zaman karmaşıklığına sahiptir?
A) $O(n)$
B) $O(\log n)$
C) $O(1)$
D) $O(n^2)$
- Bir string dizisinin hash tablosuna eklenmesi nasıl yapılır?
- Hash tablosunda bir elemanın aranması nasıl gerçekleşir?
- Farklı anahtarlar için aynı hash değerinin üretilmesi durumunda ne olur?
- Tanımla: Hash Tablosu Nedir?
- Tanımla: Karma Fonksiyonu Nedir?
- Tanımla: Çakışma (Collision) Nedir?

Cevap Anahtarı

1. B) Anahtarları benzersiz dizinlere dönüştürmek — Karma fonksiyonunun ana görevi, anahtarları kullanarak verimli bir şekilde erişilebilecek benzersiz dizinler oluşturmaktır.
2. C) Veriyi Silmek — Veriyi silmek çakışma çözme yöntemi değildir. Zincirleme, Açık Adresleme ve Doğrusal Sondalama (Açık Adresleme'nin bir türü) çakışma çözme yöntemleridir.
3. C) $O(1)$ — İdeal bir hash tablosunda, çakışmaların az olduğu veya iyi yönetildiği durumlarda, ortalama erişim zaman karmaşıklığı $O(1)$ 'dir.
4. 1. Eklenicecek string'in karma fonksiyonu ile bir hash değeri hesaplanır. 2. Hash değerinin mod tablosunun boyutu alınarak bir indeks bulunur. 3. Eğer o indekste eleman yoksa string eklenir. 4. Eğer eleman varsa çakışma çözülür (örneğin zincirleme ile) ve string eklenir.
5. 1. Aranacak elemanın anahtarının karma fonksiyonu ile bir hash değeri hesaplanır. 2. Hash değerinin mod tablosunun boyutu alınarak bir indeks bulunur. 3. Eğer o indekste aranan eleman varsa bulunur. 4. Eğer eleman yoksa veya çakışma varsa çakışma çözme yöntemiyle arama devam eder.
6. Bu duruma çakışma (collision) denir. Çakışmaları çözmek için farklı yöntemler kullanılır: 1. Zincirleme (Chaining): Her indekste bir bağlı liste tutulur. 2. Açık Adresleme (Open Addressing): Çakışma durumunda farklı bir boş indeks aranır.
7. Anahtar-değer çiftlerini depolamak için kullanılan, karma fonksiyonları aracılığıyla anahtarları dizinlere dönüştüren bir veri yapısıdır.
8. Herhangi bir boyuttaki veriyi sabit boyutlu bir çıktıya (hash değeri) dönüştüren fonksiyondur.
9. Farklı anahtarların aynı hash değerini üretmesi durumudur.

Bounlu

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