

Rabin-Karp Algoritması Nedir?

Çalışma Kağıdı

Rabin-Karp, desen ve metin arasındaki eşleşmeleri kontrol etmek için özet (hash) fonksiyonlarını kullanan bir dizge arama algoritmasıdır. Kaydırma penceresi mantığıyla çalışarak olası eşleşmeleri hızlıca tespit eder.

$$h(s) = (s[0] * p^{(m-1)} + s[1] * p^{(m-2)} + \dots + s[m-1] * p^0) \text{ pmoc}$$

$h(s)$ = Dize s 'nin özeti (hash değeri)

s = İncelenen dize (desen veya metin parçası)

p = Bir asal sayı (genellikle alfabetik karakter sayısı)

m = Desen uzunluğu

q = Büyük bir asal sayı (modül)

Sorular

1. Rabin-Karp algoritması hangi yöntemi kullanır?

- A) Yalnızca karakter karşılaştırması
- B) Özet (hash) fonksiyonları ve kaydırma penceresi
- C) Ağaç yapıları
- D) Graf algoritmaları

2. Özet çakışması durumunda ne yapılmalıdır?

- A) Algoritma durdurulur
- B) Karakterler birebir karşılaştırılır
- C) Özet fonksiyonu değiştirilir
- D) Metin yeniden aranır

3. Rabin-Karp algoritmasının en kötü durum zaman karmaşıklığı nedir?

- A) $O(n)$
- B) $O(m)$
- C) $O(nm)$
- D) $O(n \log m)$

4. Metin: 'ABABDABACDABABCABAB', Desen: 'ABABC'

5. Rabin-Karp'ın temel avantajı nedir?

6. Tanımla: Rabin-Karp'ın ana fikri nedir?

7. Tanımla: Hangi problem için kullanılır?

8. Tanımla: Kaydırma penceresi ne anlama gelir?

Cevap Anahtarı

1. B) Özet (hash) fonksiyonları ve kaydırma penceresi — Rabin-Karp, metin ve desenin özet değerlerini karşılaştırarak çalışır ve bu karşılaştırmalar için kaydırma penceresi mantığını kullanır.
2. B) Karakterler birebir karşılaştırılır — Özet çakışması olduğunda, bu durumun gerçek bir eşleşme mi yoksa tesadüfi bir çakışma mı olduğunu anlamak için desen ile metin parçasının karakterleri tek tek karşılaştırılmalıdır.
3. C) $O(nm)$ — En kötü durumda, her özet karşılaştırması bir karakter karşılaştırmasına yol açarsa (örneğin, çok sayıda özet çakışması olduğunda), karmaşıklık $O(nm)$ olabilir.
4. 1. Desen 'ABABC' için bir özet (hash) değeri hesaplanır. 2. Metnin ilk 5 karakteri olan 'ABABD' için özet hesaplanır. 3. Eğer özetler eşleşirse, karakterler birebir karşılaştırılır. 4. Özetler eşleşmezse veya karakterler uyuşmazsa, metin penceresi bir kaydırılır ve adım 2'den devam edilir.
5. Bu işlem metnin sonuna kadar tekrarlanır.
5. Temel avantajı, özellikle çok sayıda desen arandığında veya desen uzunluğu büyük olduğunda, ortalama durumda hızlı olmasıdır. Özet fonksiyonu sayesinde tam karakter karşılaştırması yapmadan birçok olası eşleşmeyi eleyebilir.
6. Özet (hash) fonksiyonlarını kullanarak desen ve metin parçalarını karşılaştırmak.
7. Metin içinde desen arama (dize arama).
8. Metin üzerinde, desen uzunluğunda bir pencerenin adım adım kaydırılması.

Bounlu

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