

Uzay Karmaşıklığı Analizi Nedir?

Çalışma Kağıdı

Uzay Karmaşıklığı Analizi, bir algoritmanın belirli bir girdi boyutu için kullandığı maksimum bellek alanını ölçen teorik bir bilgisayar bilimi dalıdır.

Sorular

1. Aşağıdakilerden hangisi uzay karmaşıklığı analizinin temel amacıdır?

- A) Algoritmanın çalışma süresini ölçmek
- B) Algoritmanın bellek kullanımını ölçmek
- C) Algoritmanın doğruluğunu kanıtlamak
- D) Algoritmanın karmaşıklığını basitleştirmek

2. Bir algoritmanın uzay karmaşıklığı $O(n)$ ise, bu ne anlama gelir?

- A) Bellek kullanımı sabittir.
- B) Bellek kullanımı girdi boyutuyla orantılıdır.
- C) Bellek kullanımı logaritmiktir.
- D) Bellek kullanımı üsseldir.

3. Hangi durumda bir algoritmanın uzay karmaşıklığı genellikle $O(1)$ olur?

- A) Çok sayıda özyinelemeli çağrı yapıldığında
- B) Girdi verisini kopyalamak için yeni bir veri yapısı oluşturulduğunda
- C) Sadece sabit sayıda değişken kullanıldığında
- D) Büyük miktarda veri işlendiğinde

4. Basit bir diziye tersine çeviren algoritmanın uzay karmaşıklığı nedir?

5. Özyinelemeli bir fonksiyonun uzay karmaşıklığı nasıl belirlenir?

6. Tanımla: Uzay Karmaşıklığı Nedir?

7. Tanımla: Zaman Karmaşıklığı ile Farkı Nedir?

8. Tanımla: Sabit Uzay Karmaşıklığı ($O(1)$)

Cevap Anahtarı

1. B) Algoritmanın bellek kullanımını ölçmek — Uzay karmaşıklığı analizi, bir algoritmanın ihtiyaç duyduğu bellek miktarını inceler.
2. B) Bellek kullanımı girdi boyutuyla orantılıdır. — $O(n)$ gösterimi, algoritmanın bellek kullanımının girdi boyutu (n) arttıkça doğrusal olarak arttığını belirtir.
3. C) Sadece sabit sayıda değişken kullanıldığında — Eğer algoritma, girdi boyutundan bağımsız olarak yalnızca sabit sayıda değişken veya bellek alanı kullanıyorsa, uzay karmaşıklığı $O(1)$ olur.
4. 1. Algoritmanın girdi dizisi dışında ek bir bellek kullanıp kullanmadığına bakılır. 2. Eğer sadece girdi dizisi üzerinde değişiklik yapılıyorsa ve ek bir veri yapısı kullanılmıyorsa, uzay karmaşıklığı sabittir ($O(1)$). 3. Eğer ters çevrilmiş diziyi saklamak için yeni bir dizi oluşturuluyorsa, uzay karmaşıklığı girdi boyutuyla orantılıdır ($O(n)$).
5. 1. Özyinelemeli çağrılarının yığın (stack) üzerinde ne kadar yer kapladığına bakılır. 2. Her özyinelemeli çağrı, fonksiyonun parametreleri ve yerel değişkenleri için yığında yer kaplar. 3. Maksimum özyinelemeli derinlik ve her çağrının kapladığı alan, toplam uzay karmaşıklığını belirler (genellikle $O(\text{derinlik})$).
6. Bir algoritmanın çalışması için gereken bellek miktarını ölçer.
7. Zaman karmaşıklığı işlem süresini, uzay karmaşıklığı bellek kullanımını ölçer.
8. Algoritmanın bellek kullanımı girdi boyutundan bağımsızdır.

Bounlu

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