

Algoritma Analizi-Ödev 3

1. İki matris A ve B'yi başlatın.
2. A: $n \times n$ matris
3. B: $n \times n$ matris
4. C: Sonucu saklamak için $n \times n$ matris
5. Giriş: A, B
6. C'yi sıfırlarla başlatın.
7. Döngü (i = 0'dan n'ye):
8. Döngü (j = 0'dan n'ye):
9. $C[i][j] = 0$
10. Döngü (k = 0'dan n'ye):
11. Döngü (k = 0'dan n'ye):
12. $C[i][j] += A[i][k] * B[k][j]$
13. Çıkış: C (sonuç matrisi)

Algoritmanın Analizi

1.İteratif Algoritma:

Karmaşıklık Analizi:

- Bu kod, iki $n \times n$ matrisin çarpımını gerçekleştirmek için üç katlı bir döngü kullanır. Zaman karmaşıklığı $O(n^3)$ olur.
- Kod basit ve anlaşılır, ancak matrislerin boyutu büyüdükçe performans kaybı yaşanır.

2.Özyinelemeli Algoritma:

Karmaşıklık Analizi:

- Bu kod, matrisleri küçük alt matrislere bölerek recursive olarak çarpımı gerçekleştirir.
- Zaman karmaşıklığı $O(n^{\log 7})$ çünkü Strassen algoritmasının kullandığı bir yaklaşımdır.
- Recursive yaklaşım, matrislerin boyutu büyüdükçe daha verimli olabilir, ancak kod daha karmaşıktır ve çok büyük matrisler için yığın aşımına neden olabilir.

Genel olarak, iki kodun da matris çarpımını gerçekleştirdikleri aynı, ancak kullandıkları yaklaşımlar farklıdır. İteratif yaklaşım basitleştirilmiş, ancak recursive yaklaşım daha verimli olabilir. Ama, recursive yaklaşımın daha karmaşık olduğunu ve çok büyük matrisler için yığın aşımına neden olabilir.