

Big-O Gosterimi Nedir?

Calisma Kagidi

$O(f(n))$ seklinde yazilan Big-O gosterimi, bir algoritmanin maliyetinin girdi buyuklugu n ile nasil olceklendiginin ust sinirini verir. En hizlidan en yavasa: $O(1)$, $O(\log n)$, $O(n)$, $O(n \log n)$, $O(n)$ yaygin siniflardir.

Sorular

1. $O(n)$ ne anlama gelir?

- A) Sabit zaman
- B) Calisma suresi girdi buyukluguyle dogrusal artar
- C) Calisma suresi her zaman 1 saniyedir
- D) Calisma suresi ustel artar

2. Sirali bir dizide binary search'un zaman karmaşıkligi nedir?

- A) $O(n)$
- B) $O(n)$
- C) $O(\log n)$
- D) $O(1)$

3. n arttikca en hizli hangisi buyur?

- A) $O(\log n)$
- B) $O(n)$
- C) $O(n)$
- D) $O(1)$

4. Big-O neden genellikle en kotu durumu ifade eder?

- A) Hesaplama daha kolay oldugu icin
- B) Girdinin diziliminden bagimsiz bir performans tavani garanti ettigi icin
- C) En iyi durum hic gercekleşmediği icin
- D) Girdi buyuklugunu tamamen goz ardi ettigi icin

5. 8 elemanli bir listede her eleman icin bir kez calisan bir dongunun Big-O'su nedir?

6. 1024 elemanli sirali bir listede binary search en kotu durumda kac adim surer?

7. $n = 6$ elemanli bir listede, her ikisi de n kez calisan ic ice dongulerde kac islem olur?

8. Tanimla: Big-O gosterimi neyi olcer?

9. Tanimla: Yaygin Big-O siniflarini hizlidan yavasa sirala.

10. Tanimla: $O(1)$ nedir?

Cevap Anahtari

1. B) Calisma suresi girdi buyukluguyle dogrusal artar - $O(n)$, islem sayisinin n ile dogrudan ve orantili sekilde olceklendigi anlamina gelir.
2. C) $O(\log n)$ - Binary search her adimda arama alanini yariya indirir, bu da logaritmik buyume verir.
3. C) $O(n)$ - $O(n)$ karesel olarak buyur; buyuk n icin dogrusal ve logaritmik buyumeyi geride birakir.
4. B) Girdinin diziliminden bagimsiz bir performans tavani garanti ettigi icin - En kotu durum analizi, algoritmanin bu sinirdan daha kotu performans gostermeyecegini garanti eder - bu en guvenli planlama varsayimidir.
5. 1 gecis, 8 karsilastirma, eleman basina 1 islem $T(n) = n O(n)$ $n = 8$ icin yaklasik 8 islem
6. Her adimda arama alanini yariya iner: 1024 512 256 ... 1 Yarilama sayisi = $\log(1024) = 10 O(\log n)$ yaklasik 10 karsilastirma, 1024 degil
7. Dis dongu: 6 tekrar Ic dongu: her seferinde 6 tekrar Toplam = $6 \cdot 6 = 36$ islem $T(n) = n O(n)$
8. Bir algoritmanin zaman veya bellek maliyetinin, girdi buyuklugu n arttikca en kotu durumda nasil buyudugunu.
9. $O(1) < O(\log n) < O(n) < O(n \log n) < O(n^2) < O(2^n)$.
10. Sabit zaman - girdi ne kadar buyuk olursa olsun islem ayni surede tamamlanir (or. diziyi indeksle erisim).

Bounlu

Tum kartlar, adim adim cozumler ve AI hoca destegi Notek uygulamasinda.
Sinav tarihlerini Promy otomatik hatirlaticiya ceviris.