

Kod Üretimi ve Optimizasyonu Nedir?

Çalışma Kağıdı

Kod üretimi, bir problemi çözmek için algoritmaların somut bir programlama dilinde ifade edilmesidir. Optimizasyon ise bu kodu daha hızlı çalışacak, daha az bellek kullanacak veya daha az enerji harcayacak şekilde iyileştirme sürecidir.

Sorular

1. Aşağıdakilerden hangisi kod optimizasyonunun temel amaçlarından biri DEĞİLDİR?

- A) Çalışma hızını artırmak
- B) Bellek kullanımını azaltmak
- C) Kodun satır sayısını artırmak
- D) Enerji tüketimini azaltmak

2. İyi bir kod üretimi için aşağıdakilerden hangisi önemlidir?

- A) Sadece hızlı çalışması
- B) Okunabilir ve anlaşılır olması
- C) En karmaşık yapıları kullanması
- D) Az sayıda yorum içermesi

3. Kod optimizasyonu hangi aşamada yapılır?

- A) Sadece kod yazılırken
- B) Sadece kod test edilirken
- C) Kod yazıldıktan sonra ve geliştirme sürecinde
- D) Sadece kod yayınlandıktan sonra

4. Basit bir döngü optimizasyonu örneği nedir?

5. Kod üretimi aşamasında dikkat edilmesi gerekenler nelerdir?

6. Tanımla: Kod Üretimi

7. Tanımla: Kod Optimizasyonu

8. Tanımla: Verimlilik

Cevap Anahtarı

1. C) Kodun satır sayısını artırmak — Kod optimizasyonunun amacı genellikle kodun daha verimli hale gelmesidir, bu da genellikle satır sayısını azaltmayı veya en azından artırmamayı hedefler.
2. B) Okunabilir ve anlaşılır olması — Okunabilirlik, kodun bakımını ve geliştirilmesini kolaylaştırdığı için iyi kod üretimi için kritik bir faktördür.
3. C) Kod yazıldıktan sonra ve geliştirme sürecinde — Optimizasyon, kod yazıldıktan sonra performans darboğazları tespit edildiğinde veya geliştirme sürecinin bir parçası olarak yapılabilir.
4. Örnek 1: Başlangıçta, bir listedeki tüm elemanların toplamını hesaplamak için basit bir döngü kullanılır. Örnek 2: Optimizasyonla, döngü yerine daha verimli bir fonksiyon (örneğin, hazır bir 'sum' fonksiyonu) veya vektörize edilmiş işlemler kullanılabilir.
5. Okunabilirlik, modülerlik, hata yönetimi ve standartlara uygunluk gibi unsurlar kod üretimi aşamasında dikkate alınmalıdır. İyi bir kod üretimi, ilerideki optimizasyonları kolaylaştırır.
6. Algoritmaların programlama dilinde ifade edilmesi süreci.
7. Kodun performansını iyileştirme (hız, bellek kullanımı vb.) süreci.
8. Kodun kaynakları ne kadar iyi kullandığını gösteren ölçüt.

Bounlu

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