

ÖDEV 1

Son teslim tarihi: 13 Kasım 2022 (23:59:59)

Ödevin konusu: Bu ödevde asimptotik notasyonların çözümü, adım adım kodların çalışma zamanının incelenmesi, aynı işi yapan farklı algoritmaların kodlanması ve bu algoritmaların analiz edilerek çalışma zamanlarının karşılaştırılması hedeflenmektedir

Ön Bilgi

Algoritmaların analizi, problemlere ait farklı çözüm yöntemlerinin verimliliğini analiz etmek için araçlar ve metodolojiler sağlayan bir bilgisayar bilimi alanıdır. Bir algoritmanın verimliliği; ne kadar zamanda çalışmasını tamamladığı, kullandığı hafıza alanı, disk alanı vb. farklı parametrelere bağlıdır. Algoritmaların analizi birçok nedenden dolayı gereklidir ancak esas olarak performansı tahmin etmek ve aynı iş için geliştirilmiş algoritmaları karşılaştırmak için kullanılır. Ayrıca, performans için garantiler sağlar ve teorik temelin anlaşılmasına yardımcı olur.

Bir algoritmanın çalışma süresinin tam bir analizi aşağıdaki adımları içermektedir:

- Algoritmayı doğru bir şekilde uygula
- Her bir temel işlem için gereken süreyi belirle
- Temel işlemlerin yürütülme sıklığını tanımlamak için kullanılacak bilinmeyen miktarları belirle
- Programa girdi için gerçekçi bir model geliştir
- Modellenmiş girdiyi varsayarak bilinmeyen miktarları analiz et.
- Her işlem için süreyi frekansla çarparak ve ardından tüm hesaplamaları ekleyerek toplam çalışma süresini hesapla

Bu ödevde, algoritmaların zaman gereksinimlerini ölçecek ve zaman karmaşıklıklarını karşılaştıracaksınız. Bir algoritmanın zaman karmaşıklığı analizinde her bir adım için gerekli zaman girdi boyutu cinsinden hesaplanırken toplamda en büyük dereceli terim algoritmanın karmaşıklığını belirlemektedir. Aşağıdaki örneği analiz edebilirsiniz:

	Unit Cost	Times
i:=1	c1	1
sum = 0	c2	1
while (i <= n)	c3	n+1
begin		
j:=1;	c4	n
while (j <= n)	c5	n*(n+1)
begin		
sum = sum + i	c6	n*n
j:= j + 1	c7	n*n
end		

```
i:= i + 1          c8          n
end
```

Toplam maliyet = $c1 + c2 + (n+1)*c3 + n*c4 + n*(n+1)*c5 + n*n*c6 + n*n*c7 + n*c8$.
Bu algoritmanın gerçekleştirimi için gerekli zaman yaklaşık n^2 olup en kötü durum analizinde $O(n^2)$ ifadesi kullanılmaktadır.

Asipmtotik notasyonların detayları bu ödev metninde verilmemiştir. Ödevi yaparken en iyi durum ve en kötü durum analizlerini araştırıp yorumlarınızda bu analizlerden bahsetmelisiniz.

Bölüm I: Aynı işi yapan farklı algoritmaların karşılaştırılması

- (a) Insertion sort ve Merge sort algoritmalarını kullanarak klavyeden dizi boyutu alınan ve bu boyuta göre elemanları kullanıcı tarafından girilebilen bir diziyi küçükten büyüğe sıralayan programı herhangi bir programlama dilinde kodlayınız. Her bir satır için açıklama ekleyiniz. (28 puan)
- (b) Aşağıda verilen 10 elemanlı diziyi insertion sort ve merge sort algoritması kullanarak her bir algoritma için gerçekleştirimini adım adım yeni bir dizi üzerinde gösteriniz. Her bir adım için detaylı açıklamalarını yazınız. Algoritmaların sözde kodu üzerinden en kötü, en iyi ve ortalama durum analizlerini matematiksel olarak ifade edip yorumlayınız. (30 puan)

15 20 3 56 76 34 2 13 65 89 12 126

Bölüm II: Çalışma zamanı analizi

- Aşağıdaki algoritmayı adım adım analiz ederek toplam maliyeti c sabitleri ve n cinsinden belirtiniz. Big-O notasyonu cinsinden algoritmanın zaman karmaşıklığını yazınız. (15 puan)

	Unit Cost	Times
j:=n	c1	
while j >=1 do	c2	
begin		
i:=j	c3	
while i>=1 do	c4	
begin		
x:=x + 1	c5	
i:= floor (i/2)	c6	
end		
j:= floor (j/2)	c7	
end		

2 Big-O tanımını kullanarak aşağıdaki ifadeleri matematiksel olarak ispatlayınız ve bulduğunuz sabitleri (c ve n0 değerlerini) açıklayınız. (27 puan)

a 10^n is not $O(2^n)$

b $2\sqrt{n} + 6 = O(\sqrt{n})$

c $(1/2)n^2 - 3n = O(n^2)$

d $5n^2 - 3n = O(n^2)$

e $2n^2 + 10$ is $O(n^2)$.

f $2n + 5$ is $\Theta(n)$.

g $5n^2 - 3n = \Theta(n^2)$

h $n^3 + 3n = \Theta(n^3 - n^2)$

i $\log(n) = \Omega(\ln(n))$

j $\log(n) = \Theta(2^{\log \log(n)})$

Raporlama

- Kodlamasını yaptığınız algoritmaların her satırına yorum yazarak algoritmaların kodunu fonksiyonlara bölerek raporunuza ekran görüntüsü olarak eklemelisiniz.
- Uygulamalarınız için ayrı bir zip dosyası alınmayacak olup rapora aktardığınız kodlar dersi veren öğretim üyesi tarafından değerlendirilecektir. Çalışır bir uygulamayı raporunuza detaylı bir şekilde açıklamanız beklenmektedir.
- Algoritmaları zaman karmaşıklığı açısından karşılaştırmanız beklenmektedir.
- Algoritmalarınızı kodlarken ve yorumlarken kullandığınız dokümanlara veya web sayfalarına rapor içerisinde referans vermelisiniz ve dolayısıyla rapora kaynakça bölümü eklemelisiniz.

Notlar

- Raporlarınızı aşağıdaki formatta Blackboard sistemine yükleyiniz.
[ogrencinumarası-ad-soyad.pdf](#)
- Ödevinizde yorum satırları ve değişken isimlerinin anlaşılır olmadığı durumda ödeviniz notlandırılmayacaktır.
- Rapor formatı Blackboard sınıfına yüklenmiş olup bu formatın dışında rapor veren öğrencilerin raporları değerlendirmeye alınmayacaktır.

Notlandırma

Ödev 100 puan üzerinden notlandırılmakta olup her bir soruya ait puanlar parantez içerisinde verilmiştir.

Not: İyi bir rapor hazırlama en az çözümlerinizi kadar önemlidir!

Akademik Bütünlük

Ödevlerdeki tüm çalışmalar, aksi belirtilmedikçe bireysel olarak yapılmalıdır. Verilen ödevler hakkında sınıf arkadaşlarınızla tartışmanız teşvik edilir, ancak bu tartışmalar soyut bir şekilde gerçekleştirilir. Yani, belirli bir sorunun belirli bir çözümüyle ilgili tartışmalar (gerçek kodda veya sözde kodda), rapor içeriğinde sınıf arkadaşınızla aynı cümlelerin ve kodların bulunması tolere edilmeyecek ve bu öğrencilerin ödevi notlandırılmayacaktır. Kısacası, bir başkasının çalışmasını kısmen veya tamamen kendi eserinizmiş gibi teslim etmek akademik dürüstlüğün ihlali olarak değerlendirilecektir. İnternetteki her şey başka biri tarafından yazıldığından, önceki koşulun internette bulunan materyaller için de geçerli olduğunu lütfen unutmayın. Yararlandığınız doküman ve internet adreslerini kaynakça kısmında belirttiğiniz durumda ve birebir cümle alıntısı yapmadığınız takdirde akademik dürüstlüğe aykırı bir davranışta bulunmamış olacaksınız.