

2021-2022 Güz Yarıyılı

Algoritma Analizi

Ödev – 2

Ders Yürütücüleri
Doç. Dr. M. Elif KARSLIGİL
Doç. Dr. M. Amaç GÜVENSAN

Konu: Dinamik Programlama

Problem: Bir web sitesinde reklamlarını yayınlatmak isteyen N adet reklam verenin reklamın yayınlanması için tercih ettikleri başlangıç zamanı, reklamın süresi ve kaç lira ödeyecekleri giriş bilgisi olarak alınıyor. Her reklam veren sadece bir reklam zamanı tercih edebiliyor. Herhangi bir zamanda sadece bir reklam yayınlanabiliyor. Bazı reklamların zamanı çakıştığına göre, **çakışmayan ve sonuçta en yüksek kazancı veren reklamları** ve **en yüksek kazancı** bulan algoritmayı dinamik programlama ile tasarlayınız.

Açıklama: Giriş bilgisi olarak her reklamın başlangıç zamanını, süresini ve ödeyeceği fiyatı alınız. Reklamlar önce **bitme zamanlarına göre** küçükten büyüğe sıralanmalıdır. İşlem sırası j. reklama geldiğinde aşağıdaki iki durumdan en kazançlısı yapılmalıdır:

- reklam_j seçilirse: toplam kazanç reklam_j'den önce olan ve reklam_j ile çakışmayan son reklamdaki toplam kazanç ile reklam_j'nin kazancının toplamına eşittir.
- reklam_j seçilmezse: toplam kazanç (j-1). adımdaki kazanç olur.

Örnek: Aşağıdaki örnekte bitme zamanlarına göre küçükten büyüğe sıralanmış 7 adet reklam için maksimum kazanç hesabı dinamik programlama yaklaşımı ile aşağıdaki gibi yapılmıştır.

j	StartTime	Duration	Value	Profit	Açıklama
1	1	3	2	2	1. reklam alınınca Profit[1]= Profit[0]+2 = 2 olur.
2	2	5	4	4	1 ve 2 çakışıyor. Profit[0]+Value[2] > Profit[1]'den iyi olduğu için Profit[2]=4 olur.
3	5	4	4	6	3'ün çakışmadığı son reklam 1, Profit[1]+Value[3]>Profit[2] olduğu için Profit[3]=6 olur.
4	2	9	7	7	4'ten önce gelen ve çakışmadığı reklam yok. Profit[0]+ Value[4] > Profit[3], Profit[4]=7 olur.
5	10	2	2	8	5'in çakışmadığı son reklam 3, Profit[3]+Value[5]>Profit[4] olduğu için Profit[5]=8 olur
6	2	13	7	8	..
7	9	7	1	8	..

Bu örnekte toplam kazanç 8, yayınlanacak reklamlar 1,3 ve 5 olur.

Ödev Teslimi:

Aşağıda verilen bütün bilgileri içeren tek bir doküman hazırlayarak **14.12.2021 saat 23:59'a** kadar online.yildiz.edu.tr üzerinden **HW2_OgrenciNumarasi.rar** dosyasını yükleyiniz.

- Ödev kontrolleriniz **15.12.2021 Çarşamba günü** dersin laboratuvar saatinde yapılacaktır. Detaylı bilgi için **Arş. Grv. Fuat ÖGME'nin** **avesis.yildiz.edu.tr** adresindeki sayfasını takip ediniz.

Teslim Edilecekler:

1. Yaptığınız çalışmayı yöntem ve uygulama bölümlerinden oluşan bir raporda anlatınız. **Yöntem** bölümünde problemi ve çözüm için önerdiğiniz yöntemi kısaca anlatıp rekürans bağıntısını yazınız.
2. Algoritmanın sıralama işlemi dahil olarak zaman ve yer karmaşıklıklarını hesaplayınız.
3. **Uygulama** bölümünde aşağıda verilen örnek için sonuçları açıklamalarınızla birlikte adım adım gösteriniz.

j	Start	Duration	Value
1	5	3	3
2	9	4	7
3	11	6	9
4	4	7	5
5	1	3	2
6	2	5	3

4. Algoritmanızın **C** dilinde programını hazırlayarak dokümana da ekleyiniz.

5. Teslim Edilecekler

- a. HW#_OgrenciNumarasi.rar (Örn: HW2_15011001.rar)
 - i. OgrenciNumarasi.pdf (Örn: 15011001.pdf)
 - ii. OgrenciNumarasi.c (Örn: 15011001.c)

Değerlendirme

Bilgisayar bilimleri anabilim dalı araştırma görevlilerinin [avesis](#) sayfalarında yer alan Ödev Teslim Kuralları başlığındaki kurallara uyulması gerekmektedir.

Algoritma Tasarımı ve Programın Çalışması: (%80)

1. Ödev, istenilen işlerin tamamını yerine getirmelidir.
2. Gereksiz kontrollerden ve işlemlerden arınmış bir tasarım yapılmalıdır.
3. Program hatasız çalışmalıdır.
4. Programın çalışması sırasında, konuyu bilmeyen kişilerin rahatlıkla anlayabilmesi için, giriş ve çıkışlarda mesajlarla bilgi verilmelidir.

Rapor Dokümantasyonu: (%20)

1. Raporun kapak sayfasında, dersin adı, öğrencinin ad, soyad ve numarası, ödev konusu bilgileri yer almalıdır.
2. Kaynak kodda değişken deklarasyonu yapılırken her değişken tek satırda tanımlanmalı, tanımın yanına değişkenin ne için kullanılacağı açıklama olarak yazılmalıdır.
3. Değişken isimleri anlamlı olmalıdır.
4. Her fonksiyonun yaptığı iş, parametreleri ve dönüş değeri açıklanmalıdır.
5. Gerekli yerlerde açıklama satırları ile kodda yapılan işlemler açıklanmalıdır.
6. Gereksiz kod tekrarı olmamalıdır.

7. Kaynak kodun formatı düzgün, okunabilir ve takip edilebilir olmalıdır.