

# Algoritma Analizi 2.ödev

**Ad:** Muhammed Hakan **Soyad:** Kılıç

**Öğrenci No:** 19011081

## Yöntem:

Tasarladığım algoritma bir web sitesinde yayınlanan, başlama süreleri, ekran süresi ve ücretleri verilen reklamlar arasından web site sahipleri için en kazançlı senaryoyu ve bu senaryo için yayınlanması gereken reklamların ne olduğunu bulma problemine çözüm getiriyor. Bu problemi çözmek için dinamik programlama tekniğinden yararlandım. Önce Sample.txt dosyasından reklam bilgilerini okuyorum. Sonra bu reklamları bitiş zamanlarına göre artan şekilde sıralıyorum. Bu sıralı diziyi “solution” adlı, dinamik programlama tekniğini kullanan fonksiyona parametre olarak yolluyorum. Tanımlayacağım tabloda satırlar reklamlara, sütunlar da saatlere tekabül edecek bu yüzden fonksiyonda önce reklam sayısı kadar satıra ve reklamlar arasında en geç bitiş süresine sahip olan reklamın bitiş süresi + 1 kadar sütuna sahip bir tam sayı matrisi oluşturuyorum. Ardından bu matrisin ilk satırını ilk reklamın bitiş süresine kadar sıfırlıyorum. Geri kalan kısımlara da bu reklamın değerini yazıyorum. Diğer reklamlar için ise o reklamın bitiş süresine kadar olan kısımları üst satırdan kopyalayıp reklamın bitiş süresi geldiğinde ise o reklamın seçilip seçilmeyeceğine karar veriyorum. Kararı o reklamdan önce gelen ve o reklamla çakışmayan reklamların kârına reklamın değerini ekleyip bir üst satırdaki değerden büyük olup olmamasına göre karar veriyorum. Karardan sonra çıkan değeri reklamın bitiş saatinin olduğu sütuna ve ondan sonraki sütunlara yazıyorum. En sonda en alt ve en son sütunda maksimum kazanç değerine ulaşıyorum. Seçilen reklamları bulmak için de matrisin en alt ve en son sütunundaki değerden başlıyorum ve eğer değer üstteki değer ile aynı ise bu reklam seçilmemiş demektir bu yüzden bir üst satıra geçiyorum. Şayet değer üstteki değerden farklı ise bu reklam seçilmiş demektir ve bu reklamın başlangıç zamanına eşit sütuna ve bir üst satıra geçiyorum ve tüm reklamlar buluna kadar aynı işlemleri tekrar ediyorum.

Algoritmamda merge sort sıralama algoritmasını kullandım. Bu algoritmanın rekürans bağıntısı,

$$C(n) = 2 \times C(n/2) + n - 1$$

Olduğundan zaman karmaşıklığı master theorem’e göre  $O(n \log n)$ ’dir ve algoritmada ekstra diziler kullandığım için yer karmaşıklığı  $O(n)$ ’dir.

Algoritmamdaki “solution” fonksiyonunun karmaşıklığını hesaplamak için aşağıdaki rekürans bağıntısını kullandım.

$$P[i,j] = \{$$

$$\max \{ V_i + p[i-1][\text{finishtime}_i]$$

$$P[i-1][j]$$

$$\}$$

$$\}$$

Zaman karmaşıklığı ve yer karmaşıklığı birbirine eşit olup  $O(n * T)$ 'dir. T bitiş süresi en geç olan reklamın bitiş süresini gösterir. Matriste T kadar sütun ve n kadar satır olacağından bu karmaşıklık hesap edilmiştir

### Uygulama:

Finish Time

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Ad1	0	0	0	0	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Ad2	0	0	0	0	2	2	3	3	3	3	3	3	3	3	3	3	3	3
Ad3	0	0	0	0	2	2	3	5	5	5	5	5	5	5	5	5	5	5
Ad4	0	0	0	0	2	2	3	5	5	5	7	7	7	7	7	7	7	7
Ad5	0	0	0	0	2	2	3	5	5	5	7	7	12	12	12	12	12	12
Ad6	0	0	0	0	2	2	3	5	5	5	7	7	12	12	12	12	12	16

Sorted

Ad6 + Ad4 + Ad1 = 16

Sıralamadan önce

İlk reklam hiçbir reklamla çakışmadığı için direkt olarak yerleşti. İkinci reklam seçilirken profitTable[2-1][ikinci reklamın başlangıç saati] adresindeki değer ve ikinci reklamın değeri toplandı ve 3 değeri bulundu. bir üstteki 2 değerinden büyük olduğu için ilgili adrese ve son sütuna kadar 3 yazıldı. Üçüncü reklam seçilirken profitTable[3-1][üçüncü reklamın başlangıç saati] ile üçüncü reklamın değeri toplandı ve 5 değeri bulundu.  $5 > 3$  olduğu için reklam seçildi ve ikinci reklam bırakıldı. Dördüncü reklam seçilirken profitTable[4-1][dördüncü reklamın başlangıç saati] ile dördüncü reklamın değeri toplandı ve 7 bulundu. Üçüncü reklam bırakıldı ve dördüncü seçildi. Beşinci reklam seçilirken profitTable[5-1][beşinci reklamın başlangıç saati] ile beşinci reklamın değeri toplandı ve 12 bulundu. Dördüncü reklam bırakılıp yeniden birinci, üçüncü ve beşinci reklam seçildi. Altıncı ve son reklam seçilirken profitTable[6-1][altıncı reklamın başlangıç saati] ile altıncı reklamın değeri toplanıp 16 bulundu. Üçüncü ve beşinci reklam bırakılıp birinci, dördüncü ve altıncı reklam seçildi. Bu reklamların sıralamadan önceki numaraları ise 3,4 ve 5'tir.

```
max_value = 16
chosen ads are ad number 3,ad number 4,ad number 5,
-----
Process exited after 0.04825 seconds with return value 0
Press any key to continue . . .
```