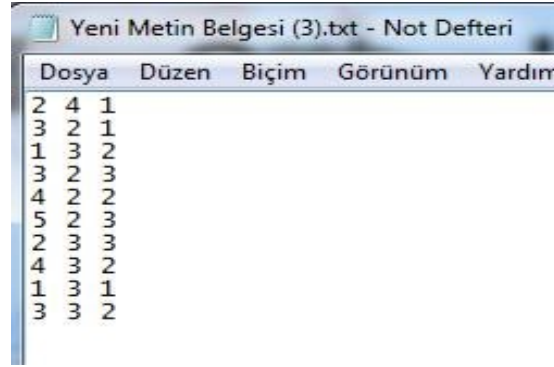


# ALGORITHMS

6

## BONUS – linked lists

Aşağıda bağlı liste yapısını oluşturacak örnek veriler gösterilmiştir.

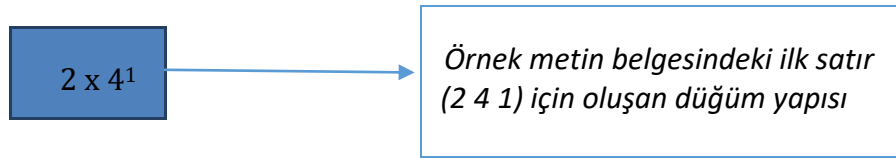


Dosya	Düzen	Biçim	Görünüm	Yardım
2	4	1		
3	2	1		
1	3	2		
3	2	3		
4	2	2		
5	2	3		
2	3	3		
4	3	2		
1	3	1		
3	3	2		

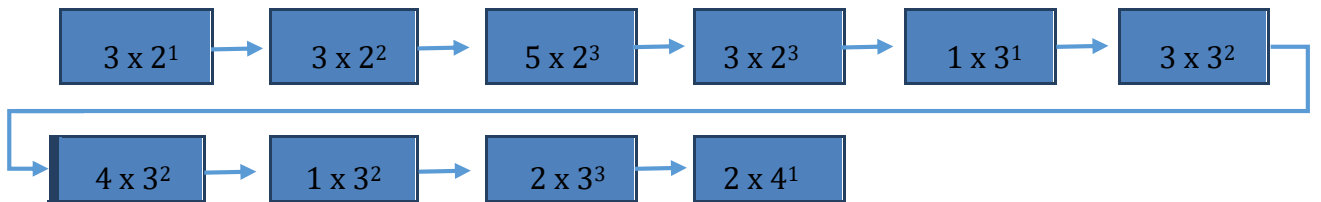
Katsayı={2,3,1,3,4,5,2,4,1,3} //1. sütun

Taban={4,2,3,2,2,2,3,3,3,3} //2.sütun

Üs={1,1,2,3,2,3,3,2,1,2} //3.sütun



**1) Ekleme işlemi:** Yukarıdaki içeriği verilen bir metin dosyasındaki katsayı, taban ve üs bilgileri tek yönlü bir bağlı listeye sıralı olarak eklenecektir. Her bir üçlü girdi (satır) aşağıdaki gibi üslü bir ifadeyi temsil etmektedir. (Burada ekleme işlemi önce taban değeri sonra üs değeri dikkate alınarak sıralı olarak yapılır) (4p)

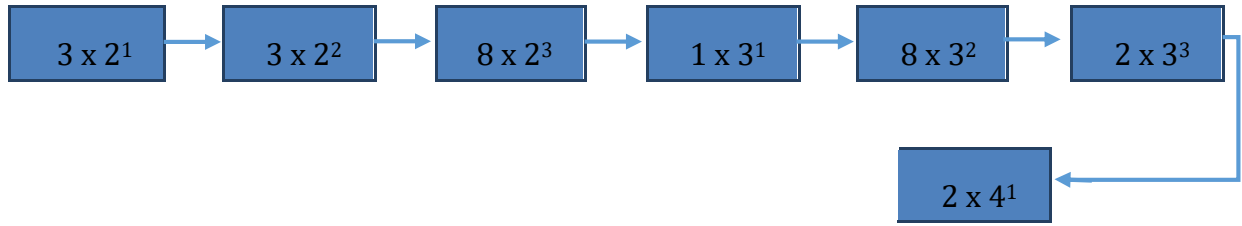


Burada; x ifadesi çarpım operatörünü ifade eder.

**2) Birleştirme işlemi:** Bağlı listedeki aynı taban ve üsse sahip olan ifadeler (düğümler) birleştirilecek ve ekranda listelenecektir. Örneğin; birleştirme (ekleme) işleminde 5x2<sup>3</sup> ve 3x2<sup>3</sup> ifadesi yerine 8x2<sup>3</sup> ifadesi alınır. Burada 5x2<sup>3</sup> ve 3x2<sup>3</sup> düğümleri silinecektir. (3p)

→ Bağlı liste yapısı kullanılarak benzer ifadeler için toplama işlemi gerçekleştirilecektir.

$$(3x2^1 + 3x2^2 + ..... + 2x4^1)$$



**3) Tümünü toplama işlemi:** Bağlı listedeki her bir düğümün değeri hesaplanarak (Örneğin:  $3x2^2 = 12$ ) bu değerler toplanacak ve sonuç ekrana yazdırılacaktır. **(3p)**

Örnek; Yukarıda verilmiş örnek girdide (*birleştirme işleminden sonraki liste*) bulunan tüm düğümlerin toplamı aşağıdaki gibi olur:

$$3x2^1 + 3x2^2 + 8x2^3 + 1x3^1 + 8x3^2 + 2x3^3 + 2x4^1 \\ = 3 + 12 + 64 + 3 + 72 + 54 + 8 = 216$$

**Not-1:** Birleştirme ve ekleme işlemi sonrasında düğümleri listeleyerek kullanıcıya göstermeyi unutmayınız.

**Not-2:** Dosyadaki veriler kullanıcı tarafından değiştirilebilir olmalıdır. Program verilerin değişebilir durumuna göre dinamik olacaktır.

Örnek gösterim:  $3x2^1 + 3x2^2 + 8x2^3 + 1x3^1 + 8x3^2 + 2x3^3 + 2x4^1$

Örnek veri yapısı:

```
struct liste {  
    int katsayı;  
    int taban;  
    int us;  
    struct Liste *sonraki; }
```