**Soru 1. [25P] a)** Yandaki algoritma A=<8,10,5,7,3,9> sayı dizisini girdi olarak aldığında ne üretir. Adım adım gösteriniz.

**b)** Bu algoritmanın çalışma zamanını (big ) cinsinden bulunuz.

**Algorithm**

**for**  **to**

**for**  **down to**

**if**

exchange with

**Soru2. [20P]** Aşağıda verilen rekürsif algoritmalar için çalışma zamanlarına ait yineleme bağıntılarını bulun ve çözün. İteratif algoritma için de çalışma zamanını *O*(büyük *O*) cinsinden ifade edin.

a) **Algorithm\_Mystery(n)**

**if** n = 1

then return 0

**else**

return Algorithm\_Mystery(n / 2) + 2

b) **Algorithm\_Noname(A,p,r)**

**if** p<r

then q=Floor[(p+r)/2]

Algorithm\_Noname(A,p,q)

Algorithm\_Noname(A,q+1,r)

c) **Algorithm\_math(A,B,n)**

**for** i = 1 **to** n

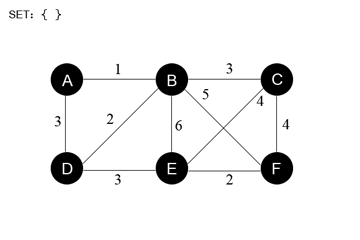
**for** j = 1 **to** n

C[i,j] = 0

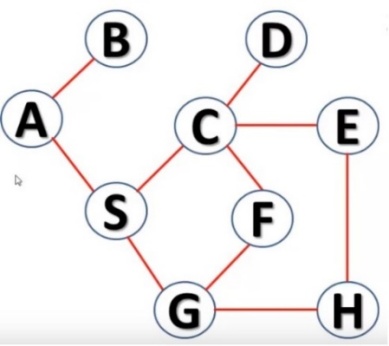
**for** k = 1 **to** n

C[i, j] = C[i,j] +A[i, k]\*B[k,j]

**Soru 3. [25P]** Kruskal Algoritmasını kullanarak, aşağıdaki grafta minimum maliyetli ağacı bulunuz.

****

**Soru 4. [25P]** A düğümünden başlayarak aşağıdaki grafı DFS (Depth First Search) yaklaşımını kullanarak geziniz ve gezintiyi şekil ile ifade ediniz. (İki düğümden birini tercih etmeniz gerekirse alfabetik sırayı kullanınız)



**Soru 5. [25P]** A=<1,1,5,3,3,3,3> sayı dizisini ele alalım. Mod(1)=2, Mod(5)=1 ve Mod(3)=4 olduğu kolaylıkla görülebilir.

Yukarıdaki örnekteki gibi, verilen bir sayı dizisindeki bir elemanın kaç kez tekrar ettiğini bulan *O*(n2) çalışma zamanına sahip bir program yazınız.