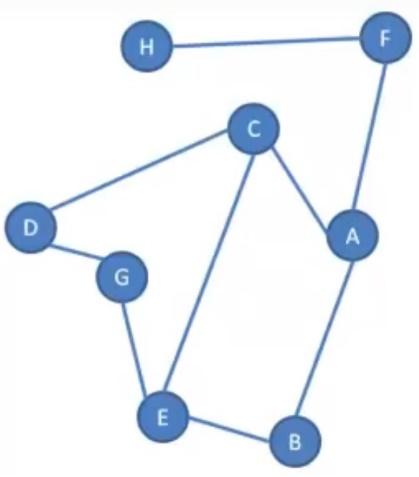
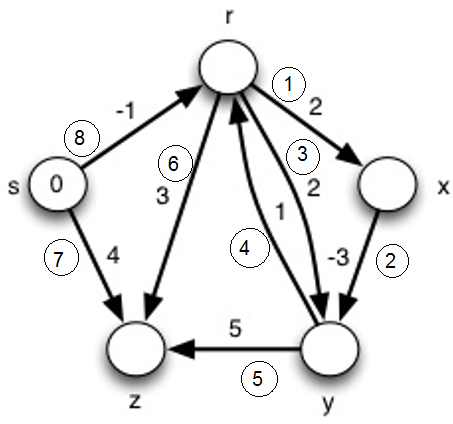
**BME222 ALGORİTMALAR FİNAL SINAVI (A) GRUBU**

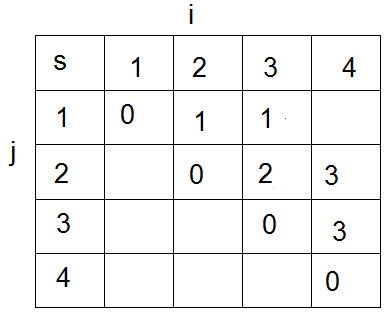
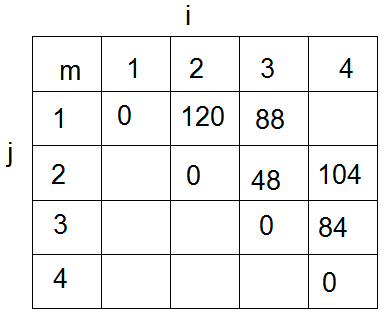
**SORU 1. (20P)** Aşağıdaki grafta, **E** düğümünden başlayarak DeSearch pth First (derinlik öncelikli arama) ve Breadth First Search (genişlik öncelikli arama) algoritmaları ile graf üzerinde dolaşın. Birden fazla tepe ele alındığında alfabetik sırada önce olanı seçip devam edin. Her iki algoritma sonucunda nasıl bir dolaşım elde edilir çiziniz? (BFS ve DFS algoritmalarının uygulanması için gerekli veri yapılarını kullanınız.)



**SORU 2. (20P)** Aşağıdaki grafta **s** kaynak tepesinden diğer tüm tepelere olan en kısa yolları ve uzunluklarını Bellman Ford Algoritmasını kullanarak adım adım bulunuz.



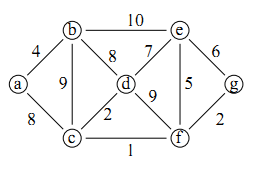
**SORU 3. (20P)** Boyutları olan matris zincir çarpımındaki optimal parantezlemeyi aşağıdaki tabloları kullanarak bulunuz.



**SORU 4. (20P)** Aşağıdaki şıklarda verilen yenileme bağıntılarını çözünüz. Çalışma zamanlarını artan sırada sıralayınız, hangisi en iyi performansa sahiptir belirtiniz.

1. T(n) = 2T(n/2)+ n2 b) T(n) = 5T(n/2)+ n2  c) T(n) = T(n/2)+T(n/4)+n

**SORU 5. (20P)** Başlangıç düğümü olarak **a** düğümünü seçtiğimizde, Prim’in algoritmasını kullanarak, aşağıdaki graftan elde edilen minimum maliyetli ağacı (MST) bulunuz. (Tablo kullanmanız tavsiye edilir)



**SORU 6. I.** Aşağıdaki ifadelerin doğru veya yanlış olduğunu belirtmek için D veya Y’yi daire içine alın. **(8P)**

1. **D Y** Bellman Ford algoritmasının zaman karmaşıklığı O(|V|+|E|) dir.
2. **D Y** Dijkstra’s Algoritması bir Böl ve Fethet Algoritması değildir.
3. **D Y** Özyinelemeli (Rekürsif) algoritmalar her zaman iteratif algoritmalardan daha iyidir.
4. **D Y** Böl ve Fethet algoritmalarında özyinelemeli prosedürler uygulanır.
5. **D Y** Ağırlıklı bir grafta kenar ağırlıklarının tamamı birbirinden farklı ise, minimum kapsayan ağaç her zaman tektir.
6. **D Y**
7. **D Y** düğümden oluşan ikili ağacın maksimum yüksekliği dir.
8. **D Y** Çabuk sıralamanın en iyi durumu her bölütleme (partition) adımında diziyi ortadan ikiye böldüğü durumdur.

**II.** Aşağıdaki ifadeleri tamamlayınız **(12P)**

1. Merge sıralamasının çalışma zamanı …………………. dır.
2. ………….. sıralamasının en kötü durum çalışma zamanı dır.
3. Strassen algoritması 8 adet özyinelemeli çarpma yerine …. adet özyinelemeli çarpma kullanır.
4. İkili arama ağaçlarında yapılan işlemlerden üç tanesi …………………………………………………..dır.
5. Dinamik programlama çözdüğü altproblemleri ……………………… saklar.
6. Negatif ağırlıklı graflarda en kısa yolları bulmak için ………………………… algoritması kullanılır.