SAKARYA ÜNİVERSİTESİ

Ayrık İşlemsel Yapılar

Hafta 13

Doç.Dr. Nilüfer YURTAY

EŞLEME (MATCHING)

13.1. Ayrık Tanıtıcılar (Distinct Representatives)

Bir çok problemin çözümü, ayrık küme elemanlarının istenen koşulu sağlayacak şekilde eşlenmesini içermektedir. Örneğin bir yaz okulu için 6 kurs açılacaktır. Tablo 13.1. de bu kursları verebilecek öğretmenler gösterilmiştir. Burada her öğretmenin verebileceği bir tek kursla görevlendirilip görevlendirilemeyeceği problemi ile ilgilenilecektir.

Tablo 13.1 Öğretmen Görevlendirmeleri

Kurs	**Öğretmenler**
1	Ayşe,Canan,Filiz
2	Canan,Deniz,Esin,Gül
3	Ayşe,Canan
4	Ayşe,Filiz
5	Banu,Esin,Gül
6	Canan,Filiz

İsimlerin baş harflerini kısaltma için kullanırsak tabloyu, Pi (i=1,2,3....6) i. kursu verebilecek öğretmenlerin kümesi olmak üzere ,

P_1= {A,C,F} , P_2 = {C,D,E,G} , P_3 = {A,C}
P_4 = {A,F} , P_5 = {B,E,G} , P_6 = {C,F}

kümeleri ile gösterebiliriz.

Eğer her bir öğretmenin sadece bir tek kursu vermesi sınırlandırması düşünülmez ise, her bir kurs için olan görevlendirme $X_1 \in P_1$, $X_2 \in P_2$, vb. olmak üzere herhangi bir (X_1,X_2, X_3,X_4, X_5,X_6) altılısı olacaktır. Bu ise P1*P2*P3*P4*P5*P6 kartezyen çarpımı sonucu elde edilir. Olası eleman sayısı 3*4*2*2*3*2 = 288'dir. İstenilen ise bu 288 altılığın içinde her biri ayrık olanlardır. Çünkü bir öğretmenin birden fazla kurs vermesi istenilmiyordu.

Tanım
S_1, S_2 .... S_n sonlu sayıda elemanı olan ve elemanlarının ayrık olması gerekmeyen kümeler olsun. S_1, S_2 .... S_n için **Ayrık Tanıtıcılar Sistemi** , $X_i \in S_i$ (i =1,2,3....n) ve tüm X_i ler ayrık olmak üzere bir (X_1,X_2, X_3,..X_n) dizisidir.

Örnek 13.1

$S_1 = \{1, 2, 3\}$, $S_2 = \{1,3\}$, $S_3 = \{1,3\}$, $S_4 = \{3,4,5\}$ kümelerinin ayrık tanıtıcı sistemleri ;

2 , 1 , 3, 4

2 , 3 ,1 , 4

2 , 1, 3, 5

2, 3, 1, 5 olacaktır.

Örnek 13.2

$S_1 = \{2, 3\}$, $S_2 = \{2,3,4,5\}$, $S_3 = \{2,3\}$, $S_4 = \{3\}$ kümeleri için ayrık tanıtıcı yoktur. Zira X_1, X_2, X_3, X_4 yapısında X_1, X_3 ve X_4 için üç ayrı eleman $S_1 \cup S_3 \cup S_4 = \{2,3\}$ olduğundan bulmak olanaksızdır.

Örnek 13.3

S = {1,2, 3, 4} için S,S,S,S dizininin 1,2,3,4 tamsayıların permütasyonu olan 4!=24 tane ayrık tanıtıcı sistemi vardır.

Teorem (Hall Teoremi)

S_1 , S_2,... S_n sonlu küme dizisinin yalnız ve yalnız { 1, 2, ... n} nin alt kümesi olan I ya karşılık gelen S_i 'lerin birleşiminin eleman sayısı en az I nın eleman sayısı kadar ise ayrık tanıtıcı sistemi vardır.

S_1 , S_2,... S_n nin sonlu elemanlı kümeler dizisi olsun. I ise {1,2,...n}'nin bir alt kümesi olsun, öyle ki bu alt kümeye karşılık gelen S_i kümelerinin birleşiminin eleman sayısı I kümesinin eleman sayısından az ise S_1 , S_2,... S_n nin ayrık tanıtıcı sistemi yoktur. Örneğimizde $S_i = P_i$ (i = 1,...6) olup I = { 1, 3, 4,6 } dır. $|\{ P1 \cup P3 \cup P4 \cup P6 \}| = 3$ olup $|I| = 4$'tür.

Örnek 13.4

S1 = { A, C , E } S2= { A , B } S3 = { B ,E } olsun. {1,2,3} ün alt kümeleri ve karşılıklı gelen S_i kümelerinin birleşimi ;

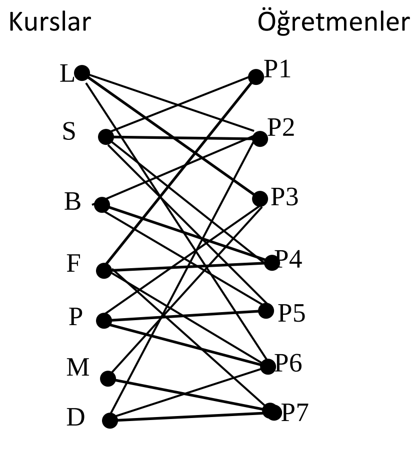
I	S_i Birleşimi
∅	∅
{1}	{ A,C,E }
{2}	{ A,B}
{3}	{ B,E}
{1,2}	{A,B,C,E}
{1,3}	{A,B,C,E}
{2,3}	{A,B,E}
{1,2,3}	{A,B,C,E}

olacaktır. Görüldüğü gibi her I alt kümesine karşılıklı gelen kümenin eleman sayısı I'nın eleman sayısının eşit yada fazla olduğuna göre S1, S2, S3 dizisinin ayrık tanıtıcı sistemi vardır. Deneme yolu ile bulunan bu örnek (A,B,E)'dir. Örnek problemimize Hall Teoremini uygularsak;

{1,2,3,4,5,6}'nın $2^6 = 64$ alt kümesi vardır. 288 olan çözümü denemekten daha iyi bir yol bulmamıza rağmen hala pratik değildir. Zira n adet S_i için 2^n adet I kümesi vardır ve n'nin büyük değerleri için 2^n çok hızlı artacaktır. Ayrıca teorem bize sadece ayrık tanıtıcıların olup olmadığını söylemekte , bunların nasıl bulunacağını söylememektedir.

13.2. Graflarda Eşleme

Eşleme problemlerinde, problem kümelerle formüle edildiğinde görülemeyen bir simetri vardır. Örneğin, her bir kursa bir öğretmen eşlemeye çalışırken tersine her bir öğretmene bir kurs eşlemek şeklinde de problemi tersine çevirebiliriz.Söz konusu olan bu simetriyi daha önce yaptığımız gibi bir graf çizerek çok daha rahat görebiliriz.(Şekil 5.2)



Şekil 13.2

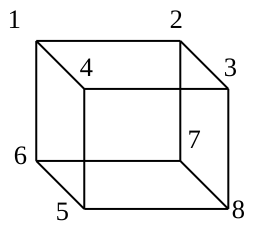
Şekilden de görüleceği gibi grafta kurstan kursa ya da öğretmenden öğretmene kenar yoktur.

Tanım

Düğüm kümesi V ve kenar kümesi ε olan bir grafta V kümesi V_1 ve V_2 gibi iki ayrık kümeye, tüm kenarlar V_1'in bir elemanından V_2'nin bir elemanına bağlı olacak şekilde ayrılabiliyorsa bu grafa **bipartite (iki-parçalı) graf** denir.

Örnek 13.5

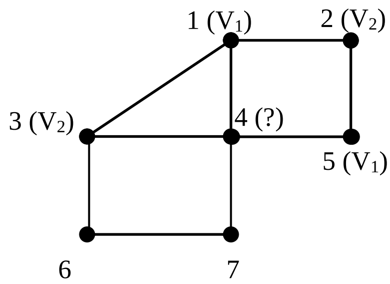
Şekil 13.3'deki graf iki parçalıdır.Çünkü tüm kenarlar tek numaralı bir düğümden çift numaralı bir düğümedir. Yani V_1 ={1,3,5,7} ve V_2 ={2,4,6,8}'dir.



Şekil 13.3

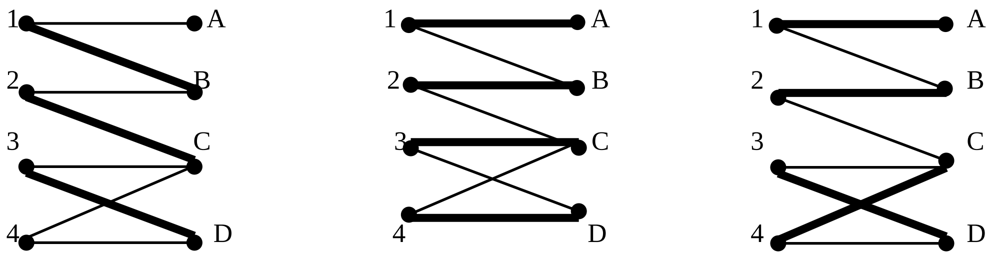
Örnek 13.6

Şekil 13.4 'deki graf ise iki parçalı değildir. Çünkü örneğin 1 düğümü V_1 de olsa 3 V_2 de olmak zorundadır. 4 etiketli düğüm ise ikisinde de olamaz.



Şekil 13.4

Eşleme problemine geri dönersek, graf gösteriminde ε kenar kümesinin bir e alt kümesini bulmak istiyoruz. Grafın hiçbir düğümü e deki kenarlardan bir tanesinden başka kenara ait olmayacak şekilde elde edilebilecek bu e alt kümesi grafın bir eşlemesi (matching) adını alacaktır. Maksimum eşleme ise en çok kenar içeren eşlemedir. Şekil 13.5' de aynı graf için olan eşlemeler belirtilmiştir.



(a) (b) (c)

eşlemede kenar sayısı 3 — kenar sayısı 4 — kenar sayısı 4

Şekil 13.5

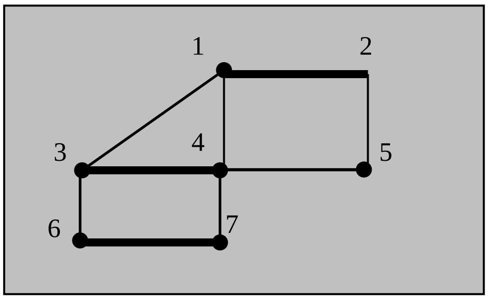
Şekilde de görüldüğü gibi ilk eşleme maksimum eşleme değildir. Ancak başka hiç bir kenar eşlenemez ise yine bir eşlemedir. Şekil 13.5(b),(c) iki ayrı eşleme daha olup, maksimum eşlemedirler. Görüldüğü gibi maksimum eşlemenin tek olması gerekmemektedir. Ayrıca eşleme tanımına bakarsak grafın iki parçalı olması gerekmemektedir. İki – parçalı grafta maksimum eşlemeyi bulmak daha kolaydır. Ancak bir çok problemde iki – parçalı olmayan graflarda eşleme istenebilir.

Örnek 13.7

7 eleman, aşağıdaki tabloda verilen yabancı dilleri konuşabilmektedir. Aynı dilde konuşabilen iki kişilik gruplar yapmak istiyoruz.

Eleman	**Diller**
1	Fransızca, Almanca, İngilizce
2	İspanyolca, Fransızca
3	Almanca, İtalyanca
4	Yunanca, Almanca, Rusça, Arapça
5	İspanyolca, Rusça
6	Çince, Japonca, İtalyanca
7	Yunanca, Çince

Elemanları bir düğüm ile gösterirsek, kenarlar iki eleman arasında ortak dil varlığını temsil ettiğinde Şekil 13.6'daki graf elde edilecektir.



Şekil 13.6

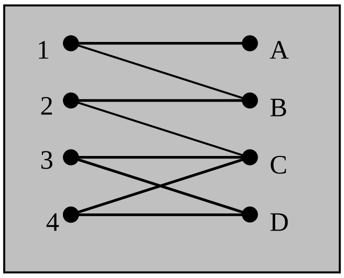
Graf iki parçaclı değildir ve bir eşleme {(1,2),(3,4),(6,7)} kenarlarıdır. Yapılan eşleme işlemi aynı zamanda maksimum eşlemedir. Zira 7 elemandan sadece 1 tanesi eşlenmemiştir.

10.3 İki-Parçalı Graf Matrisi

İki-parçalı grafı matrisle göstermek istersek, matrisin satırları V_1 kümesinin elemanlarına, sütunları da V_2 kümesinin elemanlarına karşılık düşürebiliriz. Bu durumda matris elemanları da eğer karşılık düşen düğümler arasında kenar varsa 1 aksi halde 0 olacaktır. Örneğin Şekil 13.7 deki grafın matris gösterimi

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} A & B & C & D \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \end{array}$$

olacaktır.



Şekil 13.7

Eşleme problemi, matris gösteriminde aynı satır ve sütunlarda olmayan 1 lerin kümesini bulmaya dönüşür. Matrisler için kullanılan termonoloji ise ayrıdır. Matrisin bir hattı (line) terimi ile satır ve sütunu ima etmekteyiz.

Tanım

Bir A matrisinde aynı hatta olmayan elemanlar kümesine **bağımsızdır (independent)** ve eğer A matrisinde örneğin 1 elemanlarının bağımsız setleri içerisinde en çok elemanı olan kümeye de **1 elemanlarının maksimum bağımsızlık kümesi** denir.

1'lerin bağımsız setine ilişkin olan elemanları (*) ile işaretleyeceğiz. Yukarıda verilen matris için olan bağımsız setler,

$$\begin{bmatrix} 1 & 1^* & 0 & 0 \\ 0 & 1 & 1^* & 0 \\ 0 & 0 & 1 & 1^* \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 1^* & 1 & 0 & 0 \\ 0 & 1^* & 1 & 0 \\ 0 & 0 & 1^* & 1 \\ 0 & 0 & 1 & 1^* \end{bmatrix} \quad \begin{bmatrix} 1^* & 1 & 0 & 0 \\ 0 & 1^* & 1 & 0 \\ 0 & 0 & 1 & 1^* \\ 0 & 0 & 1^* & 1 \end{bmatrix}$$

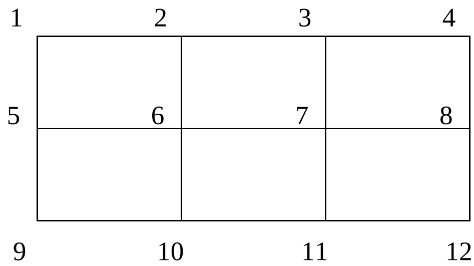
Olup Şekil 13.5' deki eşlemelere karşılık düşmektedir. Terminoloji farklı olmasına rağmen, iki-parçalı grafta maksimum eşlemeyi bulmak, matris gösteriminde 1 elemanlarının maksimum bağımsız setini bulmakla aynıdır.

13.4 Kapsama (Covering)

Bir grafın, her bir kenarının en az bir düğümünün bulunduğu bir C düğümler kümesine grafın bir ***C kapsaması (covering)*** denir. Eğer C den başka hiçbir kapsama daha az düğüm içermiyorsa C ***minimum kapsama***dır denir. Örneğin {2,3,4,5,6} Şekil 13.6 deki graf için bir kapsamadır, ancak minimum değildir. Zira {1,3,5,7} kapsaması daha az elemanlıdır.

Örnek 13.8

Şekil 13.8 daki grafta düğümler bir şehir bölgesindeki sokakların kesim noktaları olsun. Bir firma bazı köşelere büfe açmak istiyor. Ancak büfelerin, mümkün olan en az sayıda olmasını ve hiç kimsenin bir bloktan daha uzağa gitmeden büfeye ulaşabilecek şekilde yerleşmesi isteniyor.

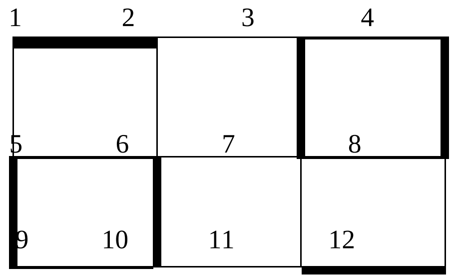


Şekil 13.8

Problemin bir kapsama problemi olduğu açıktır. Bir kapsama {1,3,6,8,9,11} dir. Bunun bir minimum kapsama olduğu ileriki bölümde gösterilecektir.

> **Teorem**
> **Bir grafın bir M eşlemesi ve C kapsaması olsun. Bu durumda $|M| \leq |C|$ dir. Ayrıca $|M|=|C|$ ise M bir maksimum eşleme ve C'de bir minimum kapsamadır.**

Son örnek için Şekil 13.9 da bir eşleme gösterilmiştir.



Şekil 13.9

{1,3,6,8,9,11}kapsama altı elemanlı ve şekil 13.8'deki eşlemede 6 elemanlıdır. O halde teoremin ikinci kısmı gereği maksimum eşleme ve minimum kapsama elde edilmiştir.

İki parçalı graf durumunda kapsama tanımını matris gösterilimi için uyarlarsak 0 ve 1lerden oluşmuş bir matrisin, 1 elemanlarının bir kapsaması matrisin tüm 1!lerini içeren hatları (satır ve/veya sutunları) kümesidir. Eğer daha az sayıda hatlı bir kapsama yok ise bu durumda minimum kapsama olacaktır.

Örnek 13.9

$$\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{bmatrix} 0 & 1^* & 0 & 0 & 0 \\ 0 & 1 & 1^* & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1^* & 0 & 0 & 1 & 1^* \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Yukarıdaki matrisi ele alalım. Tüm 1'lerin kapsaması 4 hat içermektedir. 3 ve 4.satırlar , 2 ve 3. sütunlar. Kapsama 4 hat içerdiğine göre , bağımsız 1'lerin kümeside en çok 4 elemanlı olacaktır. Bu durumda maksimum bağımsız set 4 elemanlı olacaktır.Bir olası set matriste (*1) ile işaretlenmiştir.

Bir Eşleme Algoritması

Şimdiye kadar ki örneklerde az eleman olduğu için deneme ile maksimum eşlemeyi bulabildik. Büyük graflar için etkin bir algoritma geliştirmemiz gerekmektedir.

Basitlik için iki-parçalı graflar için algoritmayı ele alacağız. Maksimum eşlemenin, maksimum bağımsız kümeyi bulmaya eşdeğer olmasından yararlanarak herhangi bir 1'lerin bağımsız seti ile başlayacağız. Algoritmayı bir örnek matris üzerinde açıklayalım.

Örnek matris, bir bağımsız 1'ler kümesi belirtilmiş olarak

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} A & B & C & D \end{array} \\ \begin{bmatrix} 1^* & 0 & 1 & 1 \\ 0 & 1^* & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \end{array}$$

şeklinde olsun. Algoritma iki temel işlemden oluşmaktadır; etiketleme ve tarama. Herhangi bir hat bir kez etiketlenmiş ise algoritmanın bir uygulama aşaması tekrar etiketlenemez. Aynı şey tarama içinde geçerlidir.

- Bir hat (satır/sütun) taranmadan önce etiketlenmelidir.
- Etiketlemeye # sembolü kullanarak, işaretsiz 1'leri içeren sütunları etiketleyerek başlıyoruz. Eğer böyle bir sütun yok ise, başlangıçtaki bağımsız setimiz zaten maksimum bağımsız set olacaktır.

$$\begin{array}{cc} & \begin{array}{cccc} A & B & C & D \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 1^* & 0 & 1 & 1 \\ 0 & 1^* & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \\ & \begin{array}{cccc} & & \# & \# \end{array} \end{array}$$

- Şimdi etiketlenmiş her bir sütunu işaretlenmemiş 1'ler için tarayacağız. C sütununda 1. satırda işaretsiz 1 bulunmaktadır. Ona ilişkin satıra bulunduğu sütunu belitmek için C etiketini vereceğiz.(Genel olarak satır etiketleri, sütun etiketleri de # etiketleri hariç satır isimleridir.) C sütununda başka 1 olmadığı için # etiketine tekrarlandığını belirten √ işareti koyup , bir sonraki 3 işaretli sütunla yani D sütununu taramaya geçiyoruz. Bu sütunda da 1. satırda 1 var ancak 1.satır önceden etiketlendiği için bir değişiklik yapmadan tarama sona erecektir. Matris,

$$\begin{array}{ccc} & \begin{array}{cccc} A & B & C & D \end{array} & \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 1^* & 0 & 1 & 1 \\ 0 & 1^* & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} & \begin{array}{c} C \\ \\ \\ \\ \end{array} \\ & \begin{array}{cccc} & & \# & \# \\ & & \surd & \surd \end{array} & \end{array}$$

görünümünde olup etiketlenmiş tüm sütunların tarama işlemi bitmiştir.

- Şimdi etiketlenmiş olan satırlar taranacaktır. Satır taramada etiketli ve taranmamış ilk satırdan başlayarak 1* varlığı sınanacaktır. Eğer bir satırda 1* bulunuyorsa bulunduğu sütun ait olduğu satır numarası (adı) ile etiketlenecek ve o satırın taranması sona erdiği için satır etiketinin yanına √ işareti konacaktır. Örneğimiz 1.satır C etiketli idi. Tarama da A sütunu 1* olduğu için A sütun etiketi 1.(satır no) olacaktır.

$$\begin{array}{ccc} & \begin{array}{cccc} A & B & C & D \end{array} & \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 1^* & 0 & 1 & 1 \\ 0 & 1^* & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} & \begin{array}{c} C\surd \\ \\ \\ \\ \end{array} \\ & \begin{array}{cccc} 1 & & \# & \# \\ & & \surd & \surd \end{array} & \end{array}$$

Satır tarama başka etiketli taranmamı satır olmadığı için sona ermiştir

- Sütun Tarama : 1 etiketli A sütunundaki 1'ler taranacak ve bulunduğu satır önceden etiketlenmemiş ise A etiketi alacaktır. Buna göre 3.satır etiketi A olacak ve tekrar satır taramaya geçilecektir.

$$\begin{array}{ccc}
 & \begin{matrix} A & B & C & D \end{matrix} & \\
\begin{matrix}1\\2\\3\\4\end{matrix} & \begin{bmatrix} 1^* & 0 & 1 & 1 \\ 0 & 1^* & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} & \begin{matrix} C\surd \\ \\ A \\ \\ \end{matrix} \\
 & \begin{matrix} 1 & & \# & \# \\ \surd & & \surd & \surd \end{matrix} &
\end{array}$$

- Satır Tarama : 3.satırda 1* aranacaktır. Görüldüğü gibi , satırda 1* yoktur. Bu durum algoritmada bir dönüm noktası olur. 3.satır A etiketli olduğu için 3.satır, A sütunundaki 1'i bir daire içine alacağız. Bu sütun 1 etiketli olduğu için 1.satırda 1*da bir daire içine alacağız. 1.saır C etiketli olduğu için C sütunundaki 1'de daire içine alınacaktır. C sütunu # ile etiketlendiği için bu işaterleme sona erecektir. Matris,

$$\begin{array}{ccc}
 & \begin{matrix} A & B & C & D \end{matrix} & \\
\begin{matrix}1\\2\\3\\4\end{matrix} & \begin{bmatrix} 1^* & 0 & 1 & 1 \\ 0 & 1^* & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} & \begin{matrix} C\surd \\ \\ A! \\ \\ \end{matrix} \\
 & \begin{matrix} 1 & & \# & \# \\ \surd & & \surd & \surd \end{matrix} &
\end{array}$$

olacaktır.

Bu noktada daha geniş bir bağımsız 1'ler setine daire içine aldığımız 1'leri 1*,1*'ları da 1 yaparak erişmiş olduk. Matris, önceki etiketlerde kaldırılarak, algoritmanın bir sonraki uygulamasına hazır edilirse,

$$\begin{array}{cc}
 & \begin{matrix} A & B & C & D \end{matrix} \\
\begin{matrix}1\\2\\3\\4\end{matrix} & \begin{bmatrix} 1 & 0 & 1^* & 1 \\ 0 & 1^* & 0 & 0 \\ 1^* & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}
\end{array}$$

olacaktır. Algoritmayı tekrar uygularsak , daha geniş bir set olmadığını göreceğiz. Bu durumda elde ettiğimiz matristeki matristeki 1*'ların kümesi bize maksimum eşlemeyi; {(3,A),(2,B),(1,C)} verecektir.

Şimdi de algoritmayı, bölümün başında ele aldığımız kurs eşleme problememize uygulayalım.

$$
\begin{array}{ccc}
 & \begin{matrix} A & B & C & D & E & F & G \end{matrix} & \\
\begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix} &
\begin{bmatrix}
1^* & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 1^* & 1 & 1 & 0 & 1 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1^* & 0 \\
0 & 1^* & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 1 & 0
\end{bmatrix} &
\begin{matrix} \\ D \\ \\ E \\ \\ \\ \end{matrix} \\
 & \begin{matrix} & & & \# & \# & & \# \\ & & & \surd & \surd & & \surd \end{matrix} &
\end{array}
$$

1) Sütun tarama-satır etiketleme sonucu

$$
\begin{array}{ccc}
 & \begin{matrix} A & B & C & D & E & F & G \end{matrix} & \\
\begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix} &
\begin{bmatrix}
1^* & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 1^* & 1 & 1 & 0 & 1 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1^* & 0 \\
0 & 1^* & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 1 & 0
\end{bmatrix} &
\begin{matrix} C\surd \\ \\ D\surd \\ C! \\ \\ E\surd \\ C \end{matrix} \\
 & \begin{matrix} 1 & 5 & 2 & \# & \# & & \# \\ & \surd & \surd & \surd & \surd & & \surd \end{matrix} &
\end{array}
$$

2) Satır tarama-Sütun etiketleme ve

3) Sütun tarama-satır etiketleme

4) Satır tarama-sütun etiketleme, 3.satırda C! genişleme var.

$$
\begin{array}{ccc}
 & \begin{matrix} A & B & C & D & E & F & G \end{matrix} & \\
\begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix} &
\begin{bmatrix}
1^* & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 1^* & 1 & 0 & 1 \\
1 & 0 & 1^* & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1^* & 0 \\
0 & 1^* & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 1 & 0
\end{bmatrix} &
\begin{matrix} E \\ \\ \\ E \\ \\ \\ \end{matrix} \\
 & \begin{matrix} & 5 & & 2 & \# & & \# \\ & \surd & & \surd & \surd & & \surd \end{matrix} &
\end{array}
$$

Buna göre maksimum eşleme {1A,5B,3C,2D,4F} olup, en başta öngördüğümüz gibi her bir kursa bir tek öğretmeni görevlendirmek olanaksız olup 6 kurs için C ya da F görevlendirilmek zorundadır.

Lemma 1

Algoritmada satır tarama işlemi sırasında 1*olmayan bir satıra gelindiğinde , 1'lerin bağımsız seti orijinalden bir fazla artacak

Lemma 2

Algoritma sonuna gelindiğinde (satır ve sütun tarama ve etiketleme işlemi bittiğinde) etiketli satırlar ve etiketlenmemiş sütunlar bir *kapsama* oluştururlar.

Lemma 3

Algoritma sonuna gelindiğinde her etiketli satır ve her etiketsiz sütun bir 1* içerir. Etiketli satır ve etiketsiz sütuna karşılık gelen eleman ise 1* değildir.Etiketli satır ve etiketsiz sütuna karşılık gelen eleman ise 1* değildir.

Teorem

Bağımsız set algoritması , maksimum olmayan bir bağımsız sete uygulanırsa eleman sayısını artırır. Eğer maksimum bağımsız sete uygulanırsa,onun maksimum olduğunu söyler.

Teorem (König Teoremi)

0 ve 1'lerden oluşmuş bir matriste 1'lerin maksimum bağımsız setinin eleman sayısı minimum kapsama ile aynı sayıda eleman içerir. İki-parçalı grafta da eşdeğer olarak maksimum eşleme minimum kaplama ile aynı sayıda eleman içerir.

Örnek 13.10

Bir iş toplantısında 6 konuşma saat 9,10,11,1,2 ve 3 olarak programlanmak istenmektedir. Bu 6 konuşmacının zaman sınırlandırılmaları ise (Ki, i=1,2..6)

K1: Sadece öğleden önce konuşabilecektir.

K2: Sadece saat 9'da veya 2'de konuşabilecektir.

K3: 9,11 ve 2'de konuşamayacaktır.

K4: Saat 2’ye kadar konuşamayacaktır.

K5:Saat 10’dan 3’e kadar konuşamayacaktır.

K6: Saat 10’dan 2’ye kadar konuşamayacaktır.

Buna göre uygun bir programın yapılıp yapılamayacağını sınamalıyız. Satırları konuşmacı sütunları saatler olarak karşılıklı düşen matriste maksimum bağımsız seti bulmalıyız. Matriste 1 elemanı konuşma yapılabileceğini gösterecektir.

$$\begin{array}{c|cccccc|l} & 9 & 10 & 11 & 1 & 2 & 3 & \\ \hline K1 & 1^* & 1 & 1 & 0 & 0 & 0 & \mathbf{11}\surd \\ K2 & 1 & 0 & 0 & 0 & 1^* & 0 & \mathbf{9}\surd \\ K3 & 0 & 1^* & 0 & 1 & 0 & 1 & 1\surd \\ K4 & 0 & 0 & 0 & 0 & 1 & 1^* & \\ K5 & 1 & 0 & 0 & 0 & 0 & 1 & 9! \\ K6 & 1 & 0 & 0 & 0 & 1 & 1 & 9 \end{array}$$

\# \#

K1 K3 √ √ K2

√ √

1.Aşama

$$\begin{array}{c|cccccc|l} & 9 & 10 & 11 & 1 & 2 & 3 & \\ \hline K1 & 1 & 1 & 1^* & 0 & 0 & 0 & \mathbf{10}\surd \\ K2 & 1 & 0 & 0 & 0 & 1^* & 0 & \\ K3 & 0 & 1^* & 0 & 1 & 0 & 1 & 1\surd \\ K4 & 0 & 0 & 0 & 0 & 1 & 1^* & \\ K5 & 1^* & 0 & 0 & 0 & 0 & 1 & \\ K6 & 1 & 0 & 0 & 0 & 1 & 1 & \end{array}$$

K3 K1 #

√ √ √

2.Aşama

Algoritma 2.aşamada sona ermiştir. Kapsamdaki eleman sayısı 5 olup, bağımsız set {(K5,9),(K3,10),)(K1,11),(K2,2),(K4,3)} ya da eşlemedeki eleman sayısı da beş olduğuna göre maksimum eşlemeye erişilmiştir.

K2 :{9,2}

K4 : {2,3}

K5 : {9,3 }

K6 : (9,2,3)

K2 U K4 U K5 U K6 = { 2,3,9,8} olup Hall teoremi gereği maksimum eşleme yapılamaz.

Bottleneck (Darboğaz) Problemi

Bir iş yerinde dört adet ayrı görev ve bunları yapabilecek 5 işçi vardır. Her bir işçinin bu işleri kaç saatte yapabileceği aşağıdaki tabloda verilmiştir.

	G1	G2	G3	G4
1	3	7	5	8
2	6	3	2	3
3	3	5	8	6
4	5	8	6	4
5	6	5	7	3

İstenen bu dört görevin mümkün olduğu kadar en kısa zamanda bitecek şekilde dört işçinin görevlendirilmesidir.

Tabloya bakıldığı zaman sadece bir tek iş 2 saatte tamamlandığı için minimum süre 2saat olamaz. 3 saat tamamlandığı için uygun olacaktır. Eşleme matrisini

0 iş süresi >3

1 iş süresi ≤3 olacak şekilde oluşturalım.

$$\begin{array}{c} \\ I1 \\ I2 \\ I3 \\ I4 \\ I5 \end{array}\begin{array}{c} \begin{array}{cccc} G1 & G2 & G3 & G4 \end{array} \\ \begin{bmatrix} 1* & 0 & 0 & 0 \\ 0 & 1* & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1* \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

Matriste kapsamayı incelersek tüm 1'ler sadece 3 hatta toplanmıştır. Biz eşleme sayısının 4 olmasını istiyoruz . König tepremi gereği çözüm yoktur.

Şimdi minimum süreyi 4 saat alarak matrisi yeniden düzenleyelim.

$$\begin{array}{c|cccc|c} & G1 & G2 & G3 & G4 & \\ \hline I1 & 1* & 0 & 0 & 0 & \\ I2 & 0 & 1* & 1 & 1 & G3 \\ I3 & 1 & 0 & 0 & 0 & \\ I4 & 0 & 0 & 0 & 1* & \\ I5 & 0 & 0 & 0 & 1 & \\ \hline & & I2 & \# & & \\ & & & \surd & & \end{array}$$

Bu matris de bağımsız seti inceleyerek çözüm olup olmadığını bakalım. Görüldüğü gibi maksimum bağımsız setin eleman sayısı 3 olup yine çözüm yoktur.

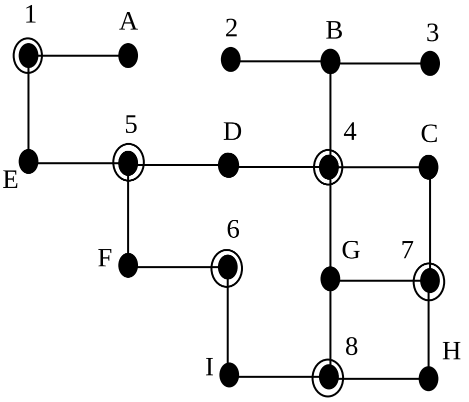
Minimum süre 5 saat olsun

$$\begin{array}{c|cccc|c} & G1 & G2 & G3 & G4 & \\ \hline I1 & 1* & 0 & 1 & 0 & G3\surd \\ I2 & 0 & 1* & 1 & 1 & G3\surd \\ I3 & 1 & 1 & 0 & 0 & G1\ ! \\ I4 & 1 & 0 & 0 & 1* & G1 \\ I5 & 0 & 1 & 0 & 1 & G2 \\ \hline & I1 & I2 & \# & & \\ & \surd & \surd & \surd & & \end{array} \Rightarrow \begin{bmatrix} 1 & 0 & 1* & 0 \\ 0 & 1* & 1 & 1 \\ 1* & 1 & 0 & 0 \\ 1 & 0 & 0 & 1* \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

Maksimum bağımsız set eleman sayısı 4'tür. En kısa zaman 5 saat olacaktır. Bir eşleme {(I1,G3),(I2,G2),(I3,G1),(I4,G4)}

Örnek 13.11

Şekil 13.10'da bir şehir haritası gösterilmektedir. Komşu düğümler bir blok mesafededir. Bu şehire bir kimsenin bir bloktan daha uzak olmaması isteği ile minimum sayıda polis noktası yerleştirilmek isteniyor. Minumum sayı kaçtır ve polisler hangi düğümlere yerleştirilecektir?



Şekil 13.10

$$\begin{array}{ccc}
 & \begin{array}{ccccccccc} A & B & C & D & E & F & G & H & I \end{array} & \\
\begin{array}{c} 1\\2\\3\\4\\5\\6\\7\\8 \end{array} &
\begin{bmatrix}
1^* & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0\\
0 & 1^* & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 1 & 1^* & 1 & 0 & 0 & 1 & 0 & 0\\
0 & 0 & 0 & 1^* & 1 & 1 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 1^* & 0 & 0 & 1\\
0 & 0 & 1 & 0 & 0 & 0 & 1^* & 1 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1^* & 1
\end{bmatrix} &
\begin{array}{l} E\surd\\ \\ \\ D\surd\\ E\surd\\ I\surd\\ H\surd\\ I\surd \end{array}\\
 & \begin{array}{ccccccccc} 1 & & 4 & 5 & \# & 6 & 7 & 8 & \# \\ \surd & & \surd & \surd & \surd & \surd & \surd & \surd & \surd \end{array} &
\end{array}$$

Algoritma sonucunda (:Lemma2) etiketlenmiş satırlar, etiketlenmemiş sütunlar kapsama oluşturur. O halde {1,4,5,6,7,8 ve B } kapsama olup minimum sayı 7 olacaktır.

13.5. Macar Yöntemi

Son örnekte 5 işçinin dört tanesi, 4 işi aynı zamanda başlayıp en kısa zamanda (5 saat) tamamlanmıştı. Çoğu zaman n adet işin toplam tamamlanma süresinin minimum yapılması istenir. Dört işçi ve dört görev için verilen tabloyu ele alarak problemin çözümünü inceleyelim.

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array}\begin{array}{c} \begin{array}{cccc} I1 & I2 & I3 & I4 \end{array} \\ \begin{bmatrix} 3 & 6 & 3 & 5 \\ 7 & 3 & 5 & 8 \\ 5 & 2 & 8 & 6 \\ 8 & 3 & 6 & 4 \end{bmatrix} \end{array}$$

Karşılıklı düzen matriste bağımsız bir eleman seti, bir olan görevlendirme olacaktır. Buna göre;

$$\begin{bmatrix} 3* & 6 & 3 & 5 \\ 7 & 3* & 5 & 8 \\ 5 & 2 & 8* & 6 \\ 8 & 3 & 6 & 4* \end{bmatrix} \qquad \begin{bmatrix} 3 & 6 & 3 & 5* \\ 7 & 3* & 5 & 8 \\ 5* & 2 & 8 & 6 \\ 8 & 3 & 6* & 4 \end{bmatrix}$$

bağımsız setleri için toplam iş sureleri 3+3+8+4=18 ve 5+3+6+5=19 saattir. İlk seçim daha iyi görünmekte ise de minimum olup olmadığını bilmiyoruz. O halde bir yöntem gerekmektedir. Söz konusu yöntem Macar yöntemidir.

Macar Algoritması :

Algoritma nxn tamsayı elemanlı bir matris için n elemanlı minimum toplamlı bağımsız seti bulur.

Adım 1 (Matrisi indirge)

a) Her bir satırın en küçük eleman değerini o satırın tüm elemanlarından çıkar.

b) Her bir sütunun en küçük eleman değerini o sütunun tüm elemanlarından çıkar.

Adım 2 (0'ların maksimum bağımsız setini bul)

Matrisin 0 elemanları için maksimum bağımsız seti (S) bul.

Adım 3 (|S|<n ise bağımsız seti genişlet)

While |S|<n

(a) Matrisin 0'larının minimum kapsamını bul.

(b) K,bu kapsama hatlarında olmayan tüm elemanların en küçüğü olsun.

(c) Kapsama hatlarında olmayan tüm elemanlardan k'yı çıkar.

(d) Kapsamadaki satır ve sütunların kesişiminde bulunan elemanlara K ekle.

(e) Yeniden bulacağın maksimum bağımsız 0'lar setini S yerine yerleştir.

Endwhile

Şimdi örneğimize algoritmayı uygulayalım.

$$\begin{bmatrix} 3 & 6 & 3 & 5 \\ 7 & 3 & 5 & 8 \\ 5 & 2 & 8 & 6 \\ 8 & 3 & 6 & 4 \end{bmatrix} \Rightarrow \underset{\text{Adım 1.a}}{\begin{bmatrix} 0 & 3 & 0 & 2 \\ 4 & 0 & 2 & 5 \\ 3 & 0 & 6 & 4 \\ 5 & 0 & 3 & 1 \end{bmatrix}} \Rightarrow \underset{\text{Adım 1.b}}{\begin{bmatrix} 0 & 3 & 0 & 1 \\ 4 & 0 & 2 & 4 \\ 3 & 0 & 6 & 3 \\ 5 & 0 & 3 & 0 \end{bmatrix}}$$

$$\underset{\text{Adım 2}}{\begin{array}{c c c}
 & \begin{matrix} A & \quad B & \quad C & \quad D \end{matrix} & \\
\begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix} & \begin{bmatrix} 0* & 3 & 0 & 1 \\ 4 & 0* & 2 & 4 \\ 3 & 0 & 6 & 3 \\ 5 & 0 & 3 & 0* \end{bmatrix} & \begin{matrix} C\surd \\ \\ \\ \\ \end{matrix} \\
 & \begin{matrix} 1 & \quad & \quad & \# & \quad \\ \surd & & & \surd & \end{matrix} &
\end{array}}
\xRightarrow[]{|S|=3<4}
\underset{\text{Adım 3.a}}{\begin{array}{c c}
\begin{bmatrix} \boxed{0* \quad 3 \quad 0 \quad 1} \\ 4 \quad 0* \quad 2 \quad 4 \\ 3 \quad 0 \quad 6 \quad 3 \\ 5 \quad 0 \quad 3 \quad 0* \end{bmatrix} & \begin{matrix} C\surd \\ \\ \\ \\ \end{matrix} \\
\begin{matrix} 1 & \quad & \quad & \# & \quad \\ \surd & & & \surd & \end{matrix} &
\end{array}}
\xRightarrow[(3.b)]{K=2}
\underset{\text{Adım 3.c,d}}{\begin{bmatrix} 0 & 5 & 0 & 3 \\ 2 & 0 & 0 & 4 \\ 1 & 0 & 4 & 3 \\ 3 & 0 & 1 & 0* \end{bmatrix}}$$

$$\underset{\text{Adım 3.e}}{\begin{array}{c c c}
 & \begin{matrix} A & \quad B & \quad C & \quad D \end{matrix} & \\
\begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix} & \begin{bmatrix} 0* & 5 & 0 & 3 \\ 2 & 0* & 0 & 4 \\ 1 & 0 & 4 & 3 \\ 3 & 0 & 1 & 0* \end{bmatrix} & \begin{matrix} C\surd \\ C\surd \\ B! \\ B \end{matrix} \\
 & \begin{matrix} 1 & 2 & \# & \quad \\ \surd & \surd & \surd & \end{matrix} &
\end{array}}
\Rightarrow
\underset{|S|=4=n \text{ son}}{\begin{bmatrix} 0* & 5 & 0 & 3 \\ 2 & 0 & 0* & 4 \\ 1 & 0* & 4 & 3 \\ 3 & 0 & 1 & 0* \end{bmatrix}}$$

$$\begin{bmatrix} 3* & 6 & 3 & 5 \\ 7 & 3 & 5* & 8 \\ 5 & 2* & 8 & 6 \\ 8 & 3 & 6 & 4* \end{bmatrix}$$

min.süre=3+2+5+4=14 saat

Kare olmayan matris için çözüm

İlk örneğimizi ele alırsak 4 görev ve beş işçi vardı. Ve tablo aşağıdaki gibiydi.

$$\begin{array}{c c}
 & \begin{matrix} I1 & I2 & I3 & I4 & I5 \end{matrix} \\
\begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix} & \begin{bmatrix} 3 & 6 & 3 & 5 & 3 \\ 7 & 3 & 5 & 8 & 5 \\ 5 & 2 & 8 & 6 & 2 \\ 8 & 3 & 6 & 4 & 4 \end{bmatrix}
\end{array}$$

Algoritma kare matrise uygulandığı için yapılacak iş elemanları 0 olan yeni bir satırı matrise eklemek olacaktır.

$$\begin{bmatrix} 3 & 6 & 3 & 5 & 3 \\ 7 & 3 & 5 & 8 & 5 \\ 5 & 2 & 8 & 6 & 2 \\ 8 & 3 & 6 & 4 & 4 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \Rightarrow \begin{array}{c|ccccc|} & A & B & C & D & E \\ \hline 1 & 0* & 3 & 0 & 2 & 0 \\ 2 & 4 & 0* & 2 & 5 & 2 \\ 3 & 3 & 0 & 6 & 4 & 0* \\ 4 & 5 & 0 & 3 & 1 & 1 \\ 5 & 0 & 0 & 0* & 0 & 0 \\ \hline & 1 & & 5 & \# & \\ & & & \surd & \surd & \end{array} \underset{K=1}{\Longrightarrow} \begin{array}{ccccc} A & B & C & D & E \\ \end{array} \begin{bmatrix} 0* & 4 & 0 & 2 & 1 \\ 4 & 0* & 1 & 4 & 2 \\ 3 & 0 & 2 & 3 & 0* \\ 5 & 0 & 2 & 0* & 1 \\ 0 & 1 & 0* & 0 & 1 \end{bmatrix}$$

|S|=5 son

$$\begin{bmatrix} 3* & 6 & 3 & 5 & 3 \\ 7 & 3* & 5 & 8 & 5 \\ 5 & 2 & 8 & 6 & 2* \\ 8 & 3 & 6 & 4* & 4 \\ 0 & 0 & 0* & 0 & 0 \end{bmatrix}$$

Minimum süre : 3+3+4+2=12 saattir.
Eşleme (I1,G1), (I2,G2) (I4,G4), (I5,G3) olacaktır.

Maksimum Toplam için Bağımsız Set

Bir kazak fabrikasında 4 işçi ve 4 makine vardır. Her bir işçinin bu makinelerde üretebildiği kazak sayısı tablodaki gibidir. Sorumuz maksimum kazak üretecek şekilde görevlendirmeyi nasıl yapmalıyız?

İşçi	M1	M2	M3	M4
1	3	6	7	4
2	4	5	5	6
3	6	3	4	4
4	5	4	3	5

Bu durumda maksimum bağımsız seti toplamı maksimum olacak şekilde bulmamız gerekiyor. Macar algoritmasını uygulamak istersek yapılacak tek iş matrisin tüm elemanlarını (–1) ile çarpıp algoritmayı aynen uygulamak olacaktır.

$$\begin{bmatrix} -3 & -6 & -7 & -4 \\ -4 & -5 & -5 & -6 \\ -6 & -3 & -4 & -4 \\ -5 & -4 & -3 & -5 \end{bmatrix} \Rightarrow \begin{bmatrix} 4 & 1 & 0 & 3 \\ 2 & 1 & 1 & 0 \\ 0 & 3 & 2 & 2 \\ 0 & 1 & 2 & 0 \end{bmatrix} \Rightarrow \begin{array}{c|cccc|c} 1 & 4 & 0* & 0 & 3 & C\surd \\ 2 & 2 & 0 & 1 & 0* & B\surd \\ 3 & 0* & 2 & 2 & 2 & \\ 4 & 0 & 0 & 2 & 0 & B! \\ \hline & & 1 & \# & 2 & \\ & & & \surd & & \end{array} \Rightarrow$$

$$\begin{bmatrix} 4 & 0 & 0* & 3 \\ 2 & 0 & 1 & 0* \\ 0* & 2 & 2 & 2 \\ 0 & 0* & 2 & 0 \end{bmatrix}$$

Maksimum kazak : 6+4+7+6=23 görevlendirme ise (I1,M3), (I2,M4), (I3,M1), (I4,M2) olacaktır.

Kaynaklar

F.Selçuk,N.Yurtay,N.Yumuşak,Ayrık İşlemsel Yapılar, Sakarya Kitabevi,2005.
İ.Kara, Olasılık, Bilim Teknik Yayınevi, Eskişehir, 2000.
"Soyut Matematik", S.Aktaş,H.Hacısalihoğlu,Z.Özel,A.Sabuncuoğlu, Gazi Ünv.Yayınları,1984,Ankara.
"Applied Combinatorics", Alan Tucker, John Wiley&Sons Inc, 1994.
"Applications of Discrete Mathematics", John G. Michaels, Kenneth H. Rosen, McGraw-Hill International Edition, 1991.
"Discrete Mathematics", Paul F. Dierker and William L.Voxman, Harcourt Brace Jovanovich International Edition, 1986.
"Discrete Mathematic and Its Applications", Kenneth H. Rosen, McGraw-Hill International Editions, 5th Edition, 1999.
"Discrete Mathematics", Richard Johnson Baugh, Prentice Hall, Fifth Edition, 2001.
"Discrete Mathematics with Graph Theory" , Edgar G. Goodaire, Michael M. Parmenter, Prentice Hall, 2nd Edition, 2001.
"Discrete Mathematics Using a Computer", Cordelia Hall and John O'Donnell, Springer, 2000.
"Discrete Mathematics with Combinatorics", James A. Anderson, Prentice Hall, 2000.
"Discrete and Combinatorial Mathematics", Ralph P. Grimaldi, Addison-Wesley, 1998.
"Discrete Mathematics", John A. Dossey, Albert D. Otto, Lawrence E. Spence, C. Vanden Eynden, Pearson Addison Wesley; 4th edition 2001.
"Essence of Discrete Mathematics", Neville Dean, Prentice Hall PTR, 1st Edition, 1996.
"Mathematics:A Discrete Introduction", Edvard R. Schneiderman, Brooks Cole; 1st edition, 2000.
"Mathematics for Computer Science", A.Arnold and I.Guessarian, Prentice Hall, 1996.
"Theory and Problems of Discrete Mathematics", Seymour Lipschuts, Marc. L. Lipson, Shaum's Outline Series, McGraw-Hill Book Company, 1997.
"2000 Solved Problems in Discrete Mathematics", Seymour Lipschuts, McGraw- Hill Trade, 1991.