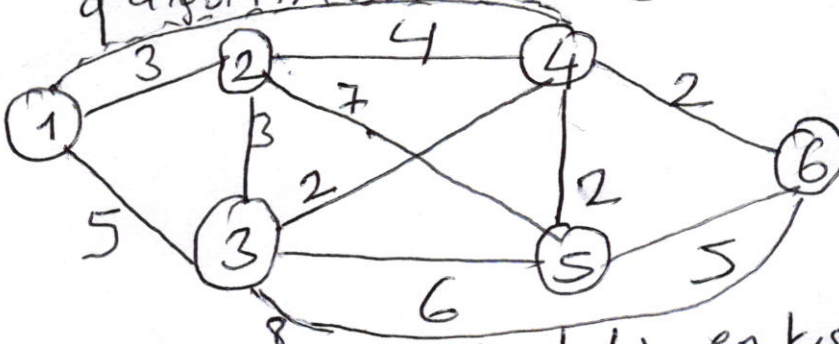


YÖNELİM ARASTIRMASI

[1]

örnek Aşağıdaki Şebekem dijital 23.10.2020
9 algoritmasını uyguluyarak Cuma 900



1 ile 6 düğümleri arasındaki en kısa mesafeyi bulunuz

Düğüm	İşaret	Durum
1	[0 -]	kalıcı
2	[3 1]	← geçici
3	[5, 1]	geçici
4	[9, 1]	geçici

1. Adım.

2. Adım	Düğüm	İşaret	Durum
	1	[0 -]	kalıcı
	2	[3, 1]	kalıcı
	3	[5, 1] yada [6, 2]	← geçici
	4	[9, 1] yada [7, 2]	geçici
	5	[10, 2] [11, 3]	geçici

2

3cü adım	<u>Değer</u>	<u>İşaret</u>	<u>Durum</u>
	1	[0 -]	kalıcı
	2	[3, 1]	kalıcı
	3	[5, 1]	kalıcı
	4	[7, 2] $\cup$ [7, 3]	$\leftarrow$ geçici
	5	[10, 2] $\cup$ [11, 3]	geçici
	6	[13, 3] $\cup$ [14, 3]	geçici

4cü Adım	<u>Değer</u>	<u>İşaret</u>	<u>Durum</u>
	1	[0 -]	kalıcı
	2	[3, 1]	kalıcı
	3	[5, 1]	kalıcı
	4	[7, 2] $\cup$ [7, 3]	kalıcı
	5	[10, 2] $\cup$ [9, 4] $\leftarrow$	geçici
	6	[13, 3] $\cup$ [9, 4]	geçici

5.ü Adım	<u>Değer</u>	<u>İşaret</u>	<u>Durum</u>
	1	[0 -]	kalıcı
	2	[3, 1]	kalıcı
	3	[5, 1]	kalıcı
	4	[7, 2] $\cup$ [7, 3]	kalıcı
	5	[9, 4]	kalıcı
	6	[9, 4] $\cup$ [14, 5]	kalıcı

6 $\rightarrow$ [9, 4] $\rightarrow$ 4 $\rightarrow$ [7, 2] $\rightarrow$ 2 $\rightarrow$ [3, 1] $\rightarrow$ 1
 $\rightarrow$ [7, 3] $\rightarrow$ 3 $\rightarrow$ [5, 1] $\rightarrow$ 1

[3] Floyd Algoritması Floyd Algoritması dijitala algoritmasından daha geneldir. Çünkü Şebebedeki herhangi iki düğüm arasındaki en kısa yolu belirler. Algoritma n düğümlü Şebekeyi n sütunlu ve n satırlı kare Matris olarak gösterir. Matrisin (i, j) elemanı i .ci düğümden j .ci düğüme olan dij. uzaklığıdır. i doğrudan j 'ye bağlı ise dij. sonlu bağlı değil ise sonsuzdur.

Floyd Algoritması Aşağıdaki şekilde i, j, k gibi üç düğüm verilmiştir. Eğer

$$d_{ij} + d_{jk} < d_{ik}$$

ise i den başlayıp j den geçerek k 'ya ulaşmak daha kısadır. Bu durumla i den k 'ya dolaylı yolu $i \rightarrow j \rightarrow k$ dolaylı yoluyla değiştirmek optimumu verir. Bu üçlü işlem değişimi aşağıdaki adımlar kullanılarak Şebekeye sistematik olarak uygulanır.



0. adım: Başlangıç uzaklık matrisi D_0 ve düğüm sırası matrisi S_0 'ı daha sonra anlaşılacağı gibi tanımla. Diyagonal elemanlar bloke edilmişlerdir göstermek için (-) ile işaretlenmiştir.

4 kst olarak belirle

$$D_0 =$$

	1	2	...	j	...	n
1	1	d_{12}	...	d_{1j}	...	d_{1n}
2	d_{21}	—	...	d_{2j}	...	d_{2n}
...						
i	d_{i1}	d_{i2}	...	d_{ij}	...	d_{in}
...						
n	d_{n1}	d_{n2}	...	d_{nj}	...	1

$$S_0 =$$

	1	2	...	j	...	n
1	—	2	...	j	...	n
2	1	—	...	j	...	n
...						
i	1	2	...	j	...	n
...						
n	1	2	...	j	...	n

k , genel adım k . Satırı ve k . sütunu anahtar satır ve anahtar sütun olarak tanımla. Tüm iyej ler için D_{k-1} deki her bir d_{ij} elemanına üçlü işlem uygula. Eğer burada

$$d_{ik} + d_{kj} < d_{ij} \quad (i \neq k, j \neq k \text{ ve } i \neq j)$$

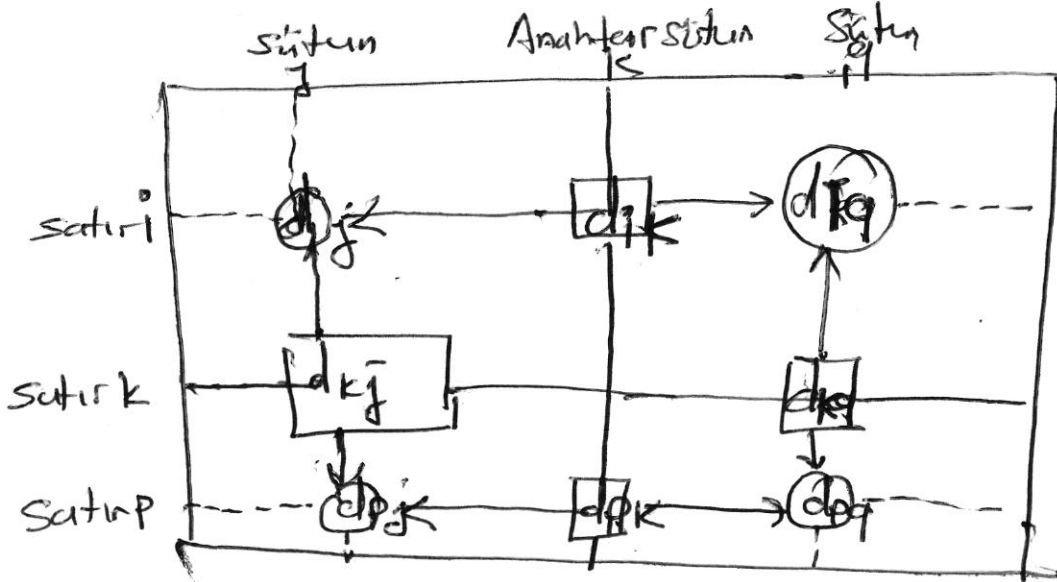
sağlanıyorsa aşağıdaki değişiklikleri yap.

(a) D_{k-1} de d_{ij} 'yi $d_{ik} + d_{kj}$ ile değiştirilerek D_k 'yi oluştur.

(b) S_{k-1} de s_{ij} 'yi k ile değiştirilerek S_k 'yi oluştur. $k = k+1$ olarak belirle ve k ci adımı tekrarla.

5

Dk-1 aşamada şekilde görüldüğü gibi gösterilerek algoritmanın k. adımı daha net bir şekilde açıklanabilir. Burada k. satır ve k. Sütun 0 andaki anahtar satır ve sütunu tanımlar. 1. 2. ..., k-1 satırlarından herhangi birini, p satırında k+1, k+2, ..., n satırlarından herhangi birini gösterebilir. Benzer şekilde j. sütunu 1, 2, ..., k-1 sütunlarından herhangi birini q satırında k+1, k+2, ..., n sütunlarından herhangi birini gösterebilir. Üçlü işlem aşağıdaki gibi uygulanır. Eğer anahtar satır ve anahtar sütundaki karelerle gösterilen elementlerin toplamı, daireyle gösterilen ilgili arakesit elementinden küçükse arakesit uzaklığı yerine anahtar uzaklıkların toplamını yazmak uygun olur.



n adım sonra i ve j düğümleri en kısa yolu Dn ve Sn matrislerinden aşağıdaki kuralları kullanarak belirleriz.

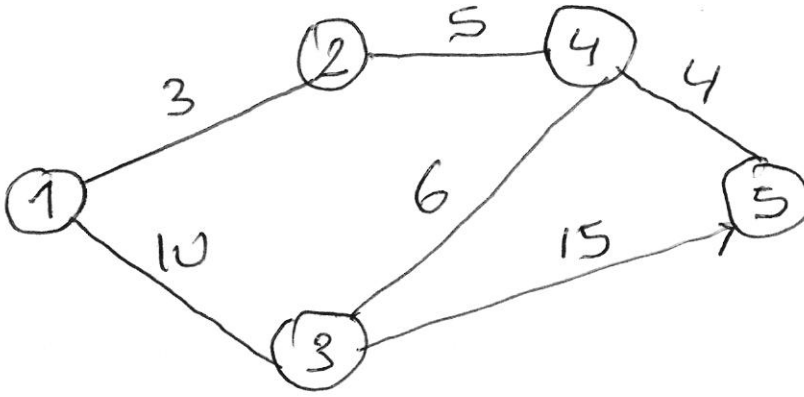
1. Dn'den dij, i ve j düğümleri arasındaki en kısa yolu verir.

2. Sn'den $i \rightarrow k \rightarrow j$ yolunu veren k-Sij ara düğümünü belirler.

6

ve $s \neq j$ ise dur, yolun tüm ara düğümleri bulunmuştur.
Aksi halde i ve k düğümleri ve k ve j düğümleri arasındaki prosedürü tekrar et.

Örnek: Aşağıdaki şebeke için iki düğüm arasındaki en kısa yolu bulun. Uzaklıklar km cinsinden belirtilmiştir. (3,5) bağlantısı 5. düğümden 3. düğüme trafiğin olmadığı tek yönlü bir bağlantıdır. Diğer düğümlerde iki yönde de trafik akışına izin verilmektedir.



0. Yineleme D_0 ve S_0 matrisleri şebekenin başlangıç durumunu verir. 5. ci düğümden 3. düğüme trafik olmadığından $d_{53} = \infty$ olmasının olışında

D_0 Simetriktr.

D_0

	1	2	3	4	5
1	-	3	10	∞	∞
2	3	-	∞	5	∞
3	10	∞	-	6	15
4	∞	5	6	-	4
5	∞	∞	∞	4	-

S_0

	1	2	3	4	5
1	-	2	3	4	5
2	1	-	3	4	5
3	1	2	-	4	5
4	1	2	3	-	5
5	1	2	3	4	-

7

1 adım: Anahtar satır ve anahtar sütun D_0 matrisinde 1 satır 1 sütun (kol olduğundan) alınır. ~~Ser~~ ~~değerler 3. düğüme trafiğe olmadığınca $d_{33} = \infty$~~ ~~olmasının dışında D_2 simetrikdir,~~

Tablo içinde kutu içine alınan d_{23} ve d_{32} üsta işlemler geliştirilmiş olan elemanlardır. Böylece D_1 ve S_1 D_0 ve S_0 dan elde etmek için

1. d_{23} 'ü $d_{21} + d_{13} = 3 + 10 = 13$ ile değiştirir. $S_{23} = 1$ olarak belirle.

2. d_{32} 'yi $d_{31} + d_{12} = 10 + 3 = 13$ ile değiştirir ve $S_{32} = 1$ olarak belirle.

Bu değişiklikler D_1 ve S_1 matrisinde gösterilmiştir.

	1	2	3	4	5
1	-	3	10	∞	∞
2	3	-	13	5	∞
3	10	13	-	6	15
4	∞	5	6	-	4
5	∞	∞	∞	4	-

	1	2	3	4	5
1	-	2	3	∞	5
2	1	-	1	4	5
3	1	1	-	4	5
4	∞	2	3	-	5
5	1	2	3	4	-

2 adım: Burada sadece 4 satır 1 sütun ile 1 satır 4. sütun elemanı değiştirilmiştir. ^{Prüf} 1 satır 4. ü sütun ilgili Anahtar satır ve anahtar sütun elemanları ile toplanarak 10, 4. ü 3 satır 1. ü sütunda 8 olarak değiştirilmiştir.

[8]

S_2 de ise 1.satır 4.sütun ile 4.satır 1.sütun
D elemanları 2ci aşamada yapıldığında 2 yazılmış

	1	2	3	4	5
1	-	3	10	8	(∞)
2	3	-	13	5	(∞)
3	10	13	-	6	15
4	8	5	6	-	4
5	∞	∞	∞	4	-

	1	2	3	4	5
1	-	2	3	2	(5)
2	1	-	1	4	(5)
3	1	1	-	4	5
4	2	2	3	-	5
5	1	2	3	4	-

3.adım: 1ci satır 5ci sütun ile 2.satır 5ci sütun.
(15+10)

değiştirilmiş 1ci satır 5ci sütun 25 ancak 5ci
satır 1ci sütun ($\infty+10=\infty$) dir. 2ci satır 5ci sütun
28 ($13+15$) olmuştur. ancak 5ci satır 2ci sütun
($13+\infty$) = ∞ değişmemiştir.

	1	2	3	4	5
1	-	3	10	8	(25)
2	3	-	13	5	(28)
3	10	13	-	6	(15)
4	8	5	6	-	4
5	(∞)	(∞)	(∞)	4	-

	1	2	3	4	5
1	-	2	3	2	(3)
2	1	-	(1)	4	(3)
3	1	1	-	4	(5)
4	2	2	3	-	5
5	(1)	(2)	(3)	4	-

4cu adım (12) 5ci satır 1.sütun, 5.satır 2.sütun (9)

2ci satır 5ci sütun 9 5ci satır 3cu sütun (10) değişmiş 1.satır 5ci sütun

1ci satır 5ci sütun 12 değiştirilmiş D_4 ve S_4 ekle edildi.

2ci satır 3cu sütun değiştirilen ilgili satır ve sütuna S_4 de
4 numarası verilmiştir

11
3cu satır 2ci sütun 11

[9]

$$D_4$$

	1	2	3	4	5
1	-	3	10	8	12
2	3	-	11	5	9
3	10	11	-	6	10
4	8	5	6	-	4
5	12	9	10	4	-

	1	2	3	4	5
1	-	2	3	2	4
2	1	-	4	4	4
3	1	4	-	4	4
4	2	2	3	-	5
5	4	4	4	4	-

k=5 için ıyileştirme söz konusu değil

$d_{15}=12$ dir yolları tayin edelim $S_{15}=7$
 ise doğrudan bağlantı var. $S_{15}=4$ $5 \neq 4$

S_{145}

$S_{14}=2$

$S_{45}=5$

↓

$S_{1245}=12$ km