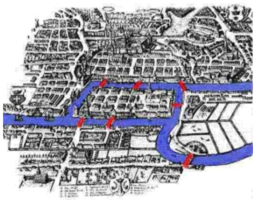
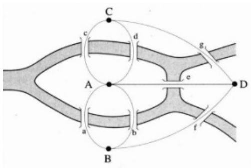
Meze

Çizge teorisi, yedi Köningsberg köprüsü boyunca güzel bir yol bulması istenen Euler ile başlamıştır



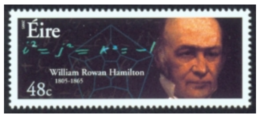
(Eulerian) yol, yedi köprünün her birinin üzerinden tam olarak bir kez geçmelidir



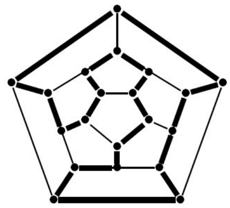
1

Meze

Bir başka erken kuş da Sir William Rowan Hamilton (1805-1865) idi



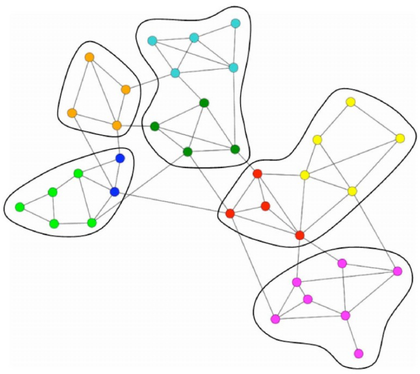
1859'da bir grafikteki tüm şehirleri tam olarak bir kez ziyaret eden bir yol bulmaya dayanan bir oyuncak geliştirdi ve bunu Dublin'deki bir oyuncak üreticisine sattı. Hiçbir zaman büyük bir başarı elde edemedi.



2

Meze

Ancak artık çizge teorisi ağlardaki toplulukları bulmak için kullanılıyor

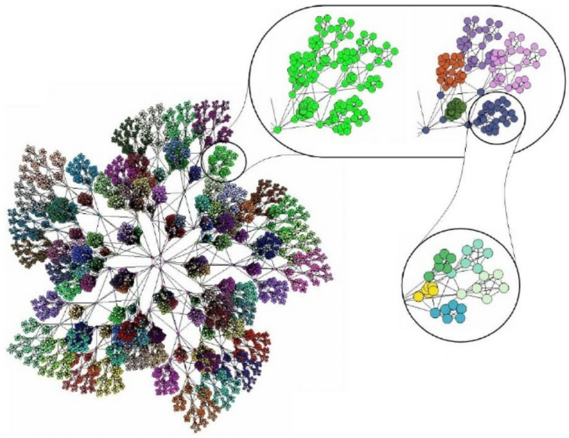


alt yapıların hiyerarşilerini tespit etmek istediğimiz yerde

3

Meze

ve boyutları oldukça büyük olabilir ...



4

Meze

Ayrıca köprüleri sıralamak (sıralamak) için de kullanılır

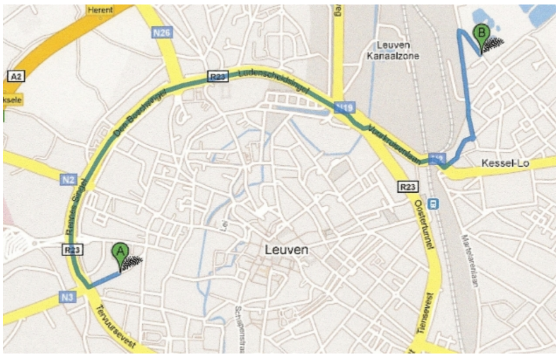
5

Meze

ya da GPS'inizle eve giden en kısa yolu bulmak için ...

6

ya da GPS'inizle eve giden en kısa yolu bulmak için ...

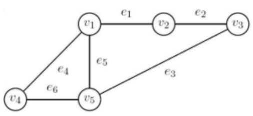


Bu kursta neleri ele alacağız

- Grafikler hakkında temel teori
  - Bağlanabilirlik
  - Yollar
  - Ağaçlar
  - Ağlar ve akışlar
  - Eulerian ve Hamiltonian grafikler
  - Renklendirme sorunları
  - Karmaşıklık sorunları

- Bir dizi uygulama (büyük grafiklerde)
  - Çizgelerde büyük ölçekli problemler
  - Büyük grafiklerde düğümlerin benzerliği
  - Telefon sorunları ve grafikleri
  - Büyük grafiklerde sıralama
  - Büyük grafiklerin kümelenmesi

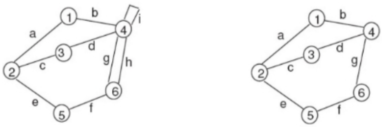
Bir $G= (V, E)$ grafiği, sonlu olduğu varsayılan bir V köşe veya düğüm) çifti ve bir E kenar kümesidir, yani $V||=n$ ve $E||=m$.



Burada $V(G)= \{v_1, v_2, . . . , v_5\}$ ve $E(G)= \{e_1, e_2, . . . , e_6\}$.

Bir $e_k= (v_{(i)}, v_{(j)})$ kenarı v_i ve v_j köşeleri ile ilişkilidir.

Basit bir grafikte aşağıdaki gibi öz döngüler veya çoklu kenarlar yoktur



Bazı özellikler

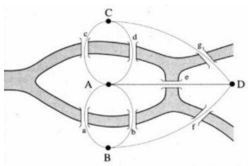
Bir V köşesinin derecesi $d(v)$, gelen kenarların sayısıdır Bir öz döngü,

derece fonksiyonunda 2 için sayılır.

Yalıtılmış bir tepe noktasının derecesi 0'dır.

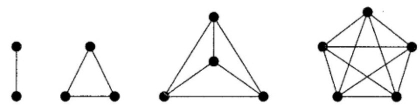
Önerme Bir G grafının derecelerinin toplamı= (V, E)
2 E||=2m'ye eşittir (önemsiz)

Sonuç Tek dereceli köşelerin sayısı çifttir (önemsizdir)



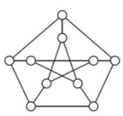
Özel grafikler

Tam bir K_n grafiği tüm $B(n, 2) := \frac{n(n-1)}{2}$ ile basit bir grafiktir.
n= 2, 3, 4, 5 için aşağıdaki matrisler gibi olası kenarlar.



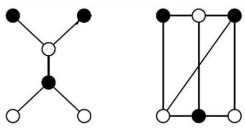
K_2 K_3 K_4 K_5

k-düzenli bir çizge, eşit k dereceli köşelere sahip basit bir çizgedir

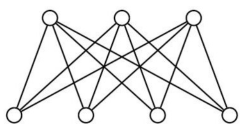


Sonuç K_n tam grafiği $(n-1)$-düzgündür

İki parçalı bir grafik, $V_{(1)}$ ve $V_{(2)}$ (aşağıdaki siyah ve beyaz düğümler) arasında hiçbir kenar olmayacak şekilde $V= V_{(1)} \cup V_{(2)}$ olan bir grafiktir



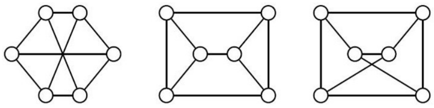
Tam bir iki parçalı çizge, V_1 ve V_2 arasındaki tüm kenarların V_2 mevcuttur (yani $E||=|V_{(1)}||V_{(2)}|$). K_{n_1,n_2} olarak not edilir.



Tam iki parçalı çizge ne zaman düzenlidir?

G ne zaman iki parçalıdır?

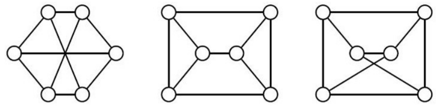
Hangi grafik iki parçalıdır?



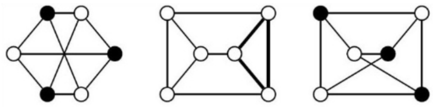
Aşağıdaki gibi kenarları ayıran 2 renk bulmak yeterlidir

G ne zaman iki parçalıdır?

Hangi grafik iki parçalıdır?



Aşağıdaki gibi kenarları ayıran 2 renk bulmak yeterlidir



İkinci örnek iki parçalı değildir çünkü bir üçgene sahiptir (devam edecek)

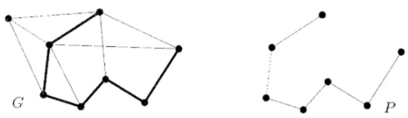
Bir grafikte yürümek

v_0 düğümünden v_k düğümüne k uzunluğunda bir yürüyüş boş olmayan bir graftır
$P=(V, E)$ formunda

$$V=\{v_0, v_1, \ldots, v_k\} \quad E=\{(v_0, v_1), \ldots, (v_{(k)-1}, v_k)\}$$

burada j kenarı − 1 ve j düğümlerini birbirine bağlar (yani V||=| E |+1). Bir iz, tüm farklı kenarlara sahip bir yürüyüştür.

Bir yol, tüm farklı düğümleri (ve dolayısıyla kenarları) içeren bir yürüyüştür.



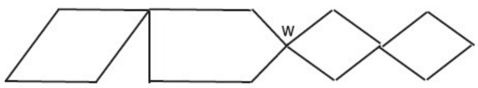
Bir yürüyüş veya patika $v_0 = v_k$ olduğunda kapalıdır.

Bir döngü, $v_0 = v_k$ dışında farklı düğümlere sahip bir yürüyüştür.

Aşağıdaki wo (yararlı) lemmaları kanıtlamaya çalışın

Önerme u'dan v'ye bir yürüyüş /= u, u'dan v'ye bir yol içerir
İpucu : alt döngüleri ortadan kaldırın

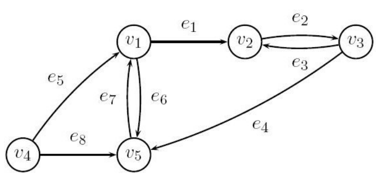
Önerme Tek uzunlukta kapalı bir yürüyüş, tek uzunlukta bir döngü içerir
İpucu : özyinelemeli olarak farklı alt grafiklere ayrıştırın ve tümevarım yöntemini kullanın



Soru Bu sadece basit grafikler için mi?

Yönlendirilmiş grafikler

Yönlendirilmiş bir grafikte veya digrafta, her kenarın bir yönü vardır.

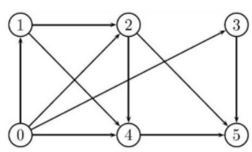


e için= $(v_{(s)}, v_{(t)})$), v_s kaynak düğüm ve v_t terminal düğümdür.

Her v düğümünün bir giriş derecesi $d_{in}(v)$ ve bir çıkış derecesi $d_{out}(v)$ vardır. Tüm düğümler için $d_{in}(v) = d_{out}(v)$ ise bir çizge dengelidir.

Topolojik düzen

Şimdi bir digrafta düğümleri sıralamaya çalışalım.



Bir f_{ord} bijeksiyonu tanımlayın : $V \to \{1, 2, \ldots, n\}$ olsun, o zaman f_{ord} (-) $G=(V, E)$ grafiği için topolojik bir sıralamadır.

$$f_{ord}(i) < f_{ord}(j), \quad \forall (i, j) \in E$$

Yukarıdaki grafik için bu mümkün görünmektedir.
Böyle bir grafiğin hiçbir döngüye sahip olmaması gerektiğini görmek kolaydır.

Ancak bu da yeterli midir?

Asiklik bir grafik, döngüleri olmayan bir grafiktir.

Önerme
Her döngüsel çizge, sıfır dereceye sahip en az bir düğüm içerir

Çelişki Kanıtı.

Tüm düğümler için $d_{in}(v) > 0$ olduğunu varsayalım, o zaman her i düğümünün $(v_{pi()}, v_{(i)})$ olacak şekilde bir öncülü $p_{(i}$) vardır. $\in E$.

Aşağıdaki gibi bir öncüller listesi oluşturmak için rastgele bir $v_{(0)}$dan başlayın

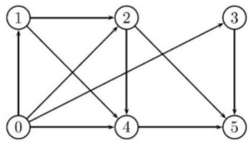
$$v_0 \to v_1 = p(v_0) \to v_2 = p(v_1)$$

| V| sınırlı olduğundan, eninde sonunda daha önce ziyaret edilmiş bir düğüme geri dönülmelidir; dolayısıyla bir döngü vardır.

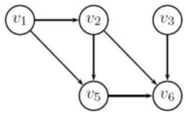
Bunu bir topolojik düzen bulmak için kullanalım

Algoritma FindTopOrd(G)
t := 0; $G^0 := G$;
while v∃∈ G^t : $d_{in}(v)$= 0 **do**
$G^{t+1} := G^{(t)} / \{v\}$; $order(v) := t+1$; $t := t+1$;
end while
eğer $t = n$ ise G asikliktir;
else if $t < n$ **then** G has a cycle; **end if end if**

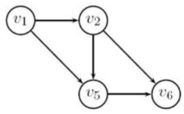
Bu algoritmayı yukarıdaki örnek üzerinde doğrulayalım.



Derecesi 0 olan tek düğüm $v_{(4)}$'tür. Dolayısıyla t= 1 için



v_4 çıkarıldıktan sonra 0. dereceden iki düğüm kalır, v_1 ve v_3. Eğer $v_{(3)}$ü seçersek, t = 2 için

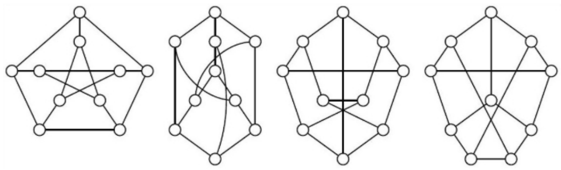


Diğer indirgemeler nihai sıralamayı $\{v_4, v_3, v_1, v_2, v_5, v_6\}$ verir. Bu

algoritmanın karmaşıklığı nedir?

İzomorfik grafikler

İki $G(_1)$ ve $G(_2)$ grafı, ilgili düğümleri arasında $G(_1)$'in her kenarının $G(_2)$'nin tam olarak bir kenarına karşılık gelmesini sağlayan bir bijeksiyon varsa izomorfiktir ve bunun tersi de geçerlidir.



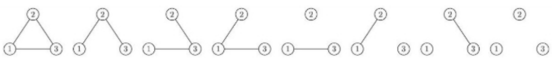
Grafikleri aynı yapan bir etiket numaralandırması bulunmalıdır Bu

problemin hala NP zor olduğuna inanılmaktadır

Sayma grafikleri

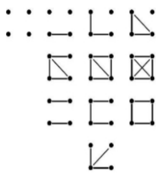
n düğümlü kaç farklı basit grafik vardır?

Düğümleri olan bir çizge $B(n, 2) := n(n-1)/2$ farklı olabilir
ve her biri mevcut olabilir veya olmayabilir.



Dolayısıyla, düğümlü en fazla $2^{n((n)-1)(/)(2)}$ çizge olabilir.
n= 3 için grafiklerin sadece 4'ü farklıdır (izomorfik olanları çıkarırsak)

n= 4 ile, izomorfik olanları daralttıktan sonra sonuçta 11 farklı grafik bulunur



n düğümlü T_n izomorfik olmayan (basit) çizge olsun. O zaman

$$L_n := \frac{2^{n(n-1)/2}}{n!} \le T_n \le 2^{n(n-1)/2}$$

Alıştırma Alt sınırı açıklayın

Logaritma almak ve $n! <^{n(n)}$ kullanmak aşağıdaki sınırları verir

$B(n, 2) - n \log n \le \log T(n) \le B(_n, 2)$ bu da

$T(n)$'in büyümesi hakkında bir fikir verir

n	2	3	4	5	6	7	8
T_n	2	4	11	34	156	1044	12346
$\lceil L_n \rceil$	2	2	3	9	46	417	6658

İzomorfizm

Bipartite yeniden ziyaret edildi

İki parçalı çizgelere tekrar bakalım

Önerme Bir çizge, tek uzunlukta döngü içermiyorsa iki parçalıdır

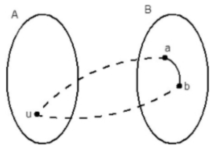
Gereklilik Önemsiz : Döngünün düğümlerini siyah ve beyaza boyayın.

Yeterlilik $u \in V$ olarak seçilsin ve $f(v)$ u'dan v'ye giden en kısa yolun uzunluğu olsun (böyle bir yol yoksa ∞)

$A = \{v \in V \mid f(v) = \texttt{tek}\}$ $B = \{v \in V \mid f(v) = \texttt{çift}\}$

O zaman A ve B, V'nin u'ya bağlı düğümlerinin bir bölümünü oluşturur.

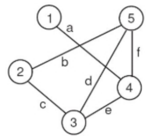
A'nın herhangi iki düğümü veya B'nin herhangi iki düğümü arasında bağlantı olmayabilir. Bu durumda, tek uzunlukta bir döngü oluşturulabilir.
Her alt grafikte tekrarlayın.

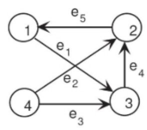


Grafikleri temsil etme

Grafikleri temsil etme

Bir $G = (V, E)$ grafiği genellikle bitişiklik matrisi ile temsil edilir. $A(i, j) = 1$ ise $(i, j) \in E$ olan bir $n \times n$ matris A'dır. Çizgeler için





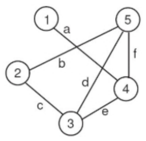
bitişiklik matrisleri şunlardır

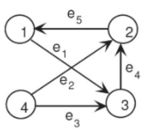
$$A_{(1} = \begin{pmatrix} 0&0&0&1&0 \\ 0&0&1&0&1 \\ 0&1&0&1&1 \\ 1&0&1&0&1 \\ 0&1&1&1&0 \end{pmatrix} \qquad A_{(2} = \begin{pmatrix} 0&0&1 \\ 1&0&0&0 \\ 0&1&0&0 \\ 0&1&1&0 \end{pmatrix}$$

Grafikleri temsil etme

Bir çizge $n \times m$ insidans matrisi T ile de temsil edilebilir. Yönlendirilmemiş

bir çizge için $T(i, k) = T(j, k) = 1$ iff $e_k = (v_i, v_j)$.
Yönlendirilmiş bir çizge için $T(i, k) = -1$; $T(j, _k) = 1$ iff $e_k = (v_i, v_j)$.
Çizgeler için



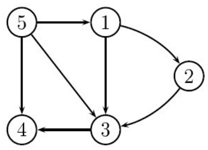


insidans matrisleri şunlardır

$$T_{(1} = \begin{pmatrix} 1&0&0&0&0&0 \\ 0&1&1&0&0&0 \\ 0&0&1&1&1&0 \\ 1&0&0&0&1&1 \\ 0&1&0&1&0&1 \end{pmatrix} \qquad T_{(2} = \begin{pmatrix} -1&0&0&0&1 \\ 0&1&0&1&-1 \\ 1&0&1&-1&0 \\ 0&-1&-1&0&0 \end{pmatrix}$$

Grafikleri temsil etme

A ve T'nin seyrek bir matris gösterimi de kullanılabilir. Bu aslında, örneğin düğümlere göre düzenlenmiş bir kenar listesinden başka bir şey değildir.



$$V(1) = \{2, 3\}$$
$$V(2) = \{3\}$$
$$V(3) = \{4\}$$
$$V(4) = \emptyset$$
$$V(5) = \{1, 3, 4\}$$

Bir grafın temsilinin bu nedenle grafikteki kenar sayısında doğrusal olduğuna dikkat edin (yani m $E = |\ |$).

Daha kesin olmak gerekirse, tüm girdileri temsil etmek için gereken bit sayısı sayılmalıdır :

$$L = (n + m) \log n$$

çünkü tepe işaretçilerini temsil etmek için log n bit gerekir.

Grafikleri temsil etme

Derece sayma

1 tüm birlerin vektörü olsun, o zaman $d_{in} = A^T \mathbf{1}$ ve $d_{out} = A\mathbf{1}$
A düğümlerinin iç derece ve dış derece vektörleridir.
ve $d_{out} = d_{in} = d$ yönlendirilmemiş grafikler için.
O halde öz döngüleri nasıl dikkate almalıyız?
Yönlendirilmemiş bir grafın bitişiklik matrisinde $A(i, i) = 2$
Yönlendirilmiş bir grafın bitişiklik matrisinde $A(i, i) = 1$

Yönlendirilmemiş bir çizge için $d = T\mathbf{1}$'e sahibiz.
Yönlendirilmiş bir çizge için T_t ve T_s matrisleri aşağıdaki gibi tanımlanabilir

terminal ve kaynak düğümlerini içeren : $T = T_t - T_s$ ile

$$T_t := \begin{pmatrix} 0&0&0&0 \\ 0&1&0&1&0 \\ 1&0&1&0&0 \\ 0&0&0&0&0 \end{pmatrix} \quad T_s := \begin{pmatrix} 0&0&0 \\ 1&1&0&0 \\ 0&0&0&1&0 \\ 0&1&1&0&0 \end{pmatrix}$$

Ayrıca $d_{in} = T_t\mathbf{1}$ ve $d_{out} = T_s\mathbf{1}$ değerlerine de sahibiz.

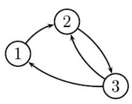
Grafikleri temsil etme

A'nın Güçleri

Önerme $(A^k)_{ij}$, i'den j'ye k uzunluğundaki yürüyüşlerin sayısıdır

Kanıt $k = 1$ için önemsiz; daha büyük *k için* tümevarım yoluyla.
$A^{k+1} = A^k \cdot$ A'nın (i, j) elemanı, aşağıdaki uzunluktaki yürüyüşlerin toplamıdır j düğümüne A bitişiklik matrisi aracılığıyla bağlı olan düğümlere k.

Aşağıdaki küçük örnek bunu doğrulamaktadır



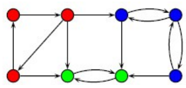
$$A = \begin{pmatrix} 0&0&1 \\ 0&1 \\ 1&1&0 \end{pmatrix}, \quad A^2 = \begin{pmatrix} 0&0 \\ 1&1&0 \\ 0&1&1 \end{pmatrix}$$

Corollary Basit bir yönlendirilmemiş grafta şu özdeşlikler vardır
$tr(A) = 0$, $tr(A^2)/2 = |E|$ ve
$tr(A^3)/6$ G'deki üçgenlerin sayısına eşittir.

Bağlı bileşenler

Yönlendirilmiş bir $G = (V, E)$ grafında, u'dan v'ye ve v'den u'ya bir yürüyüş varsa, *u ve v* güçlü bir şekilde bağlantılıdır.

Bu bir denklik ilişkisidir ve dolayısıyla *G* grafiğinin bağlantılı bileşenleri olarak adlandırılan denklik sınıflarına yol açar.



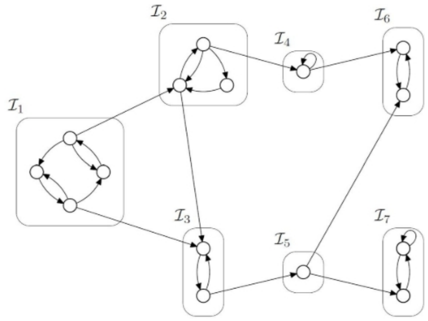
Bağlı bileşenlerine indirgenmiş çizge asikliktir (neden?)

Bu, sözlük grafiği gibi birçok uygulamada ortaya çıkmaktadır.

Bağlı bileşenler, tanımlarında birbirlerini kullanan kelime gruplarıdır (daha sonra bakınız).

İndirgeme işleminden sonra, topolojik olarak sıralanabilen bir asiklik graf elde edilir.

O zaman ne elde edersiniz? Sınıf sıralamaları

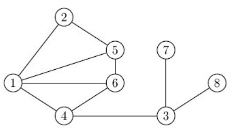


Bir başlangıç sınıfı $d_{in}(c) = 0$. Bir son sınıf $d_{out}(c) = 0$. Diğerleri ara sınıftır.

Bitişiklik listesine dayanarak bir grafiğin (güçlü) bağlanabilirliğini doğrulayın

Fikir : *s* düğümünden başlayın, grafiği keşfedin, ziyaret ettiklerinizi işaretleyin

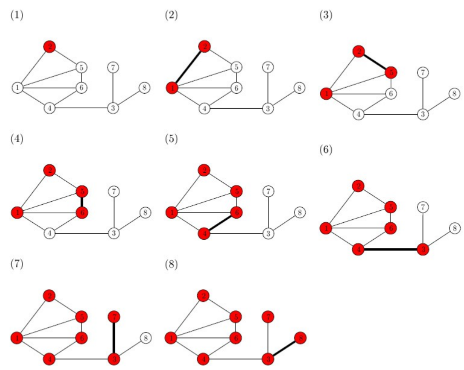
$V(1) = \{2,4,5,6\}$
$V(2) = \{1,5\}$
$V(3) = \{4,7,8\}$
$V(4) = \{1,3,6\}$
$V(5) = \{1,2,6\}$
$V(6) = \{1,4,5\}$
$V(7) = \{3\}$
$V(8) = \{3\}$



Algoritma GenericSearch(G,s) mark(*s*); *L* := $\{s\}$
while *L* / $=\emptyset$**do**
u seçin∈ *L*;
eğer∃ *(u, v)* öyle ki *v* işaretlenmemiş olsun, **o zaman**
mark(*v*); *L* := *L*∪ $\{v\}$;
başka
L := *L* \ $\{u\}$;
end if end while

Aşağıda seçilen düğümleri ve keşfedilen düğümleri işaretledik

L	
{2}	2
{2, 1}	1
{2, 1, 5}	5
{2, 1, 5, 6}	6
{1, 5, 6}	
{1, 5, 6, 4}	4
{5, 6, 4}	
{5, 4}	
{5, 4, 3}	3
{5, 3}	
{5, 3, 7}	7
{5, 3}	
{3}	
{3, 8}	8
{3}	
{}	



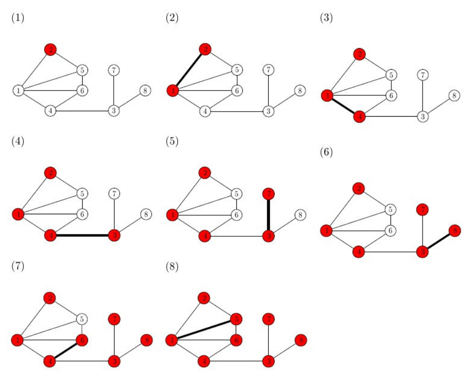
Bu algoritmanın 2n adımı vardır: her düğüm bir kez eklenir ve bir kez çıkarılır. Bu nedenle karmaşıklığı *n*'de doğrusaldır.

Seçenekler nedeniyle, bu algoritma farklı versiyonlara izin verir L için bir LIFO listesi kullanalım (Son Giren İlk Çıkar) ve *u* için *L*'ye eklenen son elemanı seçelim. Bu bir derinlik ilk aramadır (DFS).

Algoritma DeptFirstSearch(G,s)
mark(*s*); *L* := $\{s\}$;
while *L* / $=\emptyset$**do**
u := *last* (*L*)
eğer∃ *(u, v)* öyle ki *v* işaretlenmemiş , **o zaman** en küçük indeksli *v* ile (*u, v*) seçin; mark(*v* ; *L* := *L*∪ $\{v\}$;
başka
L := *L* \ $\{u\}$;
end if end while

Aşağıda seçilen düğümleri ve keşfedilen düğümleri işaretledik

L	
{2}	2
{2, 1}	1
{2, 1, 4}	4
{2, 1, 4, 3}	3
{2, 1, 4, 3, 7} 7	
{2, 1, 4, 3}	
{2, 1, 4, 3, 8} 8	
{2, 1, 4, 3}	
{2, 1, 4}	
{2, 1, 4, 6}	6
{2, 1, 4, 6, 5} 5	
{2, 1, 4, 6}	
{2, 1, 4}	
{2, 1}	
{2}	
{}	



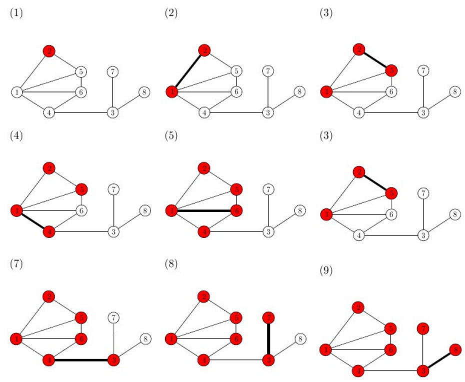
Bu algoritma, genel algoritmadan daha uzun yollar oluşturur (önce derinlik).

Şimdi *L* için bir FIFO listesi kullanıyoruz (İlk Giren İlk Çıkar) ve *u* için *L*'ye eklenen ilk eleman. Bu bir genişlik öncelikli aramadır (BFS).

Algoritma BreadthFirstSearch(G,s) mark(*s*); *L* := {*s*};
while *L* / =∅**do**
u := *first* (*L*)
eğer∃ *(u, v)* öyle ki *v* işaretlenmemiş , **o**
zaman en küçük indeksli *v* ile *(u, v)*
seçin; mark(*v* ; *L* := *L*∪ {*v*};
başka
L := *L* \ {*u*}
end if end
while

Aşağıda seçilen düğümleri ve keşfedilen düğümleri işaretledik

L	
{2}	2
{2, 1}	1
{2, 1, 5}	5
{1, 5}	
{1, 5, 4}	4
{1, 5, 4, 6} 6	
{5, 4, 6}	
{4, 6}	
{4, 6, 3}	3
{6, 3}	
{3}	
{3, 7}	
{3, 7, 8}	8
{7, 8}	
{8}	
{}	

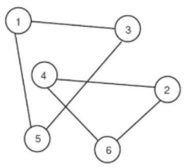


Bu algoritma daha geniş bir ağaç oluşturur (önce genişlik).

Bağlanabilirliğin test edilmesi

Keşif algoritması, belirli bir *u* düğümünden bir yol ile ulaşılabilecek tüm düğümlerin kümesini bulur∈ *V* .

Çizge yönlendirilmemişse, bu kümedeki her düğüm *u*'ya geri dönen bir yol izleyebilir. Böylece *u*'nun bağlantılı bileşeni *C*(*u*)'yu oluştururlar.



Tüm bağlantılı bileşenleri bulmak için, bu araştırmayı *V**C*(*u*)'nun bir düğümünde tekrarlayın, vb.

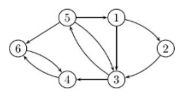
Güçlü bağlantıyı test etme

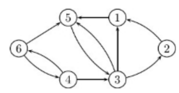
Önerme *G*= (*V, E*) bir digraf ve *u*∈ *V* olsun.
Eğer v∀∈ *V* adresinde *u*'dan *v*'ye giden bir yol ve *v*'den *u*'ya giden bir yol varsa, *G* güçlü bağlantılıdır.

Keşif algoritması, belirli bir *u* düğümünden bir yol ile ulaşılabilecek tüm düğümlerin kümesini bulur∈ *V* .

u'ya ulaşılabilecek düğümler nasıl bulunabilir? Bunun için tüm okları

tersine çevirerek ters grafiği oluşturun

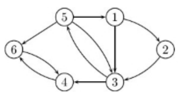


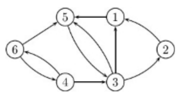


Bu grafın bitişiklik matrisinin sadece A^T olduğunu gösterin.

Önerme *G*= (*V, E*) bir digraf ve *u*∈ *V* olsun.
R_+ *(u)* *u*'dan ulaşılabilecek düğümler olsun
ve R_- *(u)* *u*'ya ulaşabilen düğümler olsun, o zaman
'nun güçlü bağlantılı bileşeni *C*(*u*) = R_+ *(u)*∩ R_- *(u)*

Ters grafiğe uygulanan keşif algoritması, *u*'dan başlayarak R_- *(u)* kümesini bulur



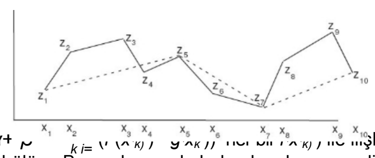


Burada R_+ *(v*(6)) = {4, 6} iken R_- *(v*(6)) = *V* dolayısıyla *C*(*v*(6)) = {4, 6} Diğer bağlı bileşenleri bulun.

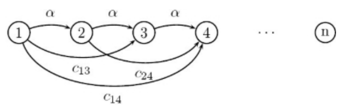
En kısa yol problemleri

Her bir kenarla ilişkilendirilmiş uzunluklara sahip yönlendirilmiş bir grafiğin iki düğümü arasındaki bir yolun en kısa toplam uzunluğunu bulun.

Örneğin, bir fonksiyonun en iyi parçalı doğrusal yaklaşımını bulun

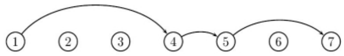


Bir maliyet $c_{ij} = \alpha + \beta \ldots_{k\,i=} (f(x_k)) - g(x_k))$ her bir $f(x_k)$ ile ilişkilidir. doğrusal bölüm. Bu, en kısa yolu bulmak anlamına gelir



Diğer örnek : Aylık talebi d_i , piyasaya sürme maliyeti $f_{(i)}$ depolama maliyeti $h_{(i)}$ ve birim fiyatı $p_{(i)}$ bir tesis için $i = 1, . . .$ her dönem için en iyi üretim politikasını bulun. , n.

Aşağıdaki yolda, örneğin 1., 4. ve 5. aşamalarda üretim yapıyoruz.

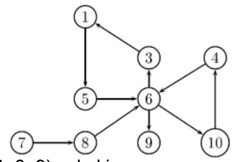


Her bölümle bir maliyet ilişkilendirilir. (1,4) yolu için bu maliyet örneğin $c_{14} = f_{(1)} + p_{(1)}(d_{(1)} + d_{(2)} + d_{(3)}) + h_{(1)}(d_{(2)} + d(3)) + h_{(2)}(d(3))$ olup sabit maliyet + 1., 2. ve 3. dönemlerdeki üretim maliyeti + 1. ve 2. dönem sonundaki depolama maliyetidir.

Toplam maliyetin minimize edilmesi, yukarıdaki gibi yolları birleştiren bir grafikte en kısa yol problemine karşılık gelir.

Önerme Eğer s'den t'ye bir en kısa yol varsa, s'den t'ye de bir en kısa yol vardır.

Kanıt
Yürüyüşün bir yol olmadığını varsayalım; dolayısıyla tekrar eden bir düğüm vardır. Bu düğümün ilk ve son oluşumu arasındaki döngüyü ortadan kaldırın. Bu prosedürü tekrarlayın.



Yukarıdaki grafikte (7, 8, 6, 3, 1, 5, 6, 10, 4, 6, 9) yolu bir döngüsü (6, 3, 1, 5, 6, 10, 4, 6). Elenmesinden sonra elimizde bir yol var (7, 8, 6, 9).

Sonuç G negatif uzunlukta döngüler içermiyorsa, ortaya çıkan yol daha düşük maliyetlidir.

Kanıt Önemsiz

Dijkstra'nın algoritması

Bu yöntem, pozitif kenar uzunluklarına sahip bir G digrafı içindir.
Yönlendirilmemiş graflar için her bir kenar aşağıdaki gibi çoğaltılabilir

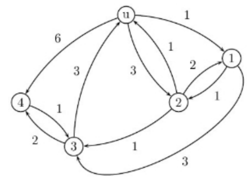
Aşağıda, $V^+(u)$ u'nun çocuklarının kümesini göstermektedir.

Algoritma Dijkstra(G,u)
$S := \{u\}$; $d(u) := 0$; $d(v) := c(u, v)\ \forall\ v\ /= u$;
while $S\ /= V$ **do**
  $v' \in / S$ seçin : $d(v') \leq d(v)\ v \forall \in / S$;
  $S := S \cup \{v'\}$;
  her v **için** $\in V^+(v')$ **do**
    $d(v) = min\{d(v), d(v') + c(v('), v)\}$
  end for end
while

Fikir : u'dan tüm en kısa yolları bildiğimiz bir S kümesini güncelleyin

Bu algoritmanın davranışını bir örnek üzerinde görelim.

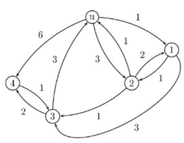
Aşağıdaki tabloda her bir düğüm için hesaplanan adımlar ve mesafeler gösterilmektedir



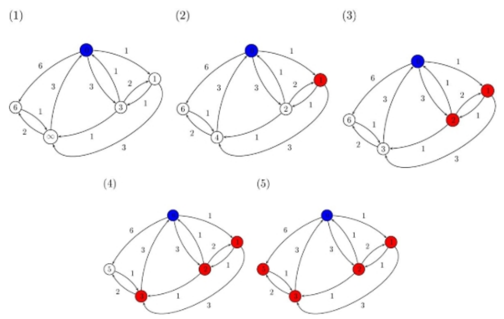
Iter	*S*	*d (u)*	d (1)	d (2)	d (3)	4
0	$\{u\}$	0	1	3	∞	6
1	$\{u, 1\}$	0	1	2	4	6
2	$\{u, 1, 2\}$	0	1	2	3	6
3	$\{u, 1, 2, 3\}$	0	1	2	3	5
4	$\{u, 1, 2, 3, 4\}$	0	1	2	3	5

Grafiğin keşfini daha ayrıntılı olarak belirtiyoruz

S	*d (u)*	d (1)	d (2)	d (3)	4
$\{u\}$	0	1	3	∞	6
$\{u, 1\}$	0	1	2	4	6
$\{u, 1, 2\}$	0	1	2	3	6
$\{u, 1, 2, 3\}$	0	1	2	3	5
$\{u, 1, 2, 3, 4\}$	0	1	2	3	5

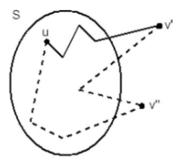


Aşağıda, u düğümü mavi ve keşfedilen düğümler kırmızıdır



Önerme Dijkstra algoritması u'dan V'nin diğer tüm düğümlerine giden en kısa yolu $O(n^2)$ zamanda bulur.

Kanıt S'nin boyutu üzerinde tümevarım yoluyla şunu gösteriyoruz
1. $\forall v \in S$, $d(v)$ u'dan v'ye giden en kısa yolun uzunluğudur
2. $\forall v \in S^+$ (S düğümlerinin çocukları), $d(v)$ u'dan v'ye giden ve yalnızca S düğümlerinden geçmeyen en kısa yolun uzunluğudur



S için önemsiz= $\{u\}$, $d(u)= 0$, $d(v) = c(u, v$.

$v(')\in / S$: $d(v(')) = \min_{(v)\in / S} d(v)$ olsun, o zaman $v(')$'ye giden en kısa yol tamamen S içinde olmalıdır. Değilse,$\exists$ $v('')$ S dışında daha kısa bir mesafede olmalıdır.

$S := S \cup \{v'\}$ şeklinde güncelleyebilir ve u'dan v''nin çocuklarına giden en kısa yolu $d(v) = min\{d(v), d(v(')) + c(v('), v)\}$ şeklinde hesaplayabiliriz. Bu, tüm $v \in S^+$ için en kısa yolun uzunluğunu verir.

Diğer mesafeler henüz bilinmemektedir ve bu nedenle∞ olarak ayarlanmıştır.

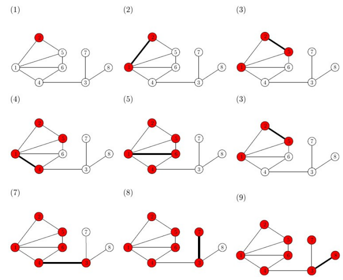
Varyantlar

Kenar uzunlukları 1 olan bir çizge için bir BFSearch yapmak ve keşif aşaması sırasında yol uzunluklarını 1 ile artırarak takip etmek yeterlidir. Dolayısıyla bu $0(m)$ zamanlı bir algoritmadır.

Algoritma ShortestPathBFS(G,v)
mark(v); S := {v}; d (v) = 0; **while**
not $S = \varnothing$ **do**
  v := *first* (S)
  eğer ∃ (v, $v^{(\prime)}$) öyle ki $v^{(\prime)}$ işaretlenmemiş olsun**, o**
   zaman en küçük indeksli $v^{(\prime)}$ ile (v, v($^{\prime}$)) seçin;
   mark($v^{(\prime)}$); $S := S \cup \{v^{(\prime)}\}$; d ($v^{(\prime)}$)= d (v)+ 1;
  başka
   $S := S \setminus \{v\}$
  end if end
while

Önerme Tam olarak k mesafesindeki tüm ilerlemeden önce doğru şekilde tanımlanır.

Kanıt k = 0 için bu önemsizdir (S orijinal u düğümüdür).
Tümevarım adımı : ifadenin k'ya kadar doğru olduğunu varsayalım.
k mesafesindeki tüm düğümler bulunduktan sonra, k'dan daha büyük bir mesafede olan düğümler bulunur, ancak hepsi komşu düğümler olduğundan, tam olarak k+ 1 mesafesinde olmalıdırlar.



Döngüsel bir çizge için, topolojik sıralama şu şekilde hesaplanabilir $O(m)$ zaman (daha önce bakınız).

En kısa yol problemini çözmek için aşağıdaki algoritma kullanılır

Algoritma ShortestPathAcyclic(G,v) d
(1)= 0; d (i) :=∞ for i= 2, . . . , n; **için** i
= 1 : n− 1 **do**
  j **için** ∈ V^{+} (i) **do**
   d (j) := min_j {d (j), d (i)+ $c(i, j$)};
  end for end
for

Bu ikinci adımın karmaşıklığı nedir?

En kısa yol problemi bir akış problemi ya da doğrusal programlama problemi olarak da görülebilir.
Bu, Bellman-Ford Algoritması gibi diğer algoritmalara yol açar.

Ağaçlar ve ormanlar

Ağaç, asiklik ve bağlantılı bir grafiktir

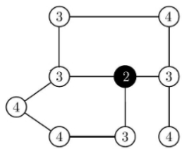
Bir orman asiklik bir graftır (ve dolayısıyla ağaçların bir birleşimidir)

Önerme n = | V | mertebesindeki bir G= (V, E) grafı için aşağıdakiler eşdeğerdir

1. G bağlantılıdır ve n− 1 kenara sahiptir
2. G asikliktir ve n− 1 kenara sahiptir
3. G bağlantılı ve asikliktir
4. ∀u, v∈ V u'dan v'ye giden bir ve yalnızca bir yol vardır
5. G asikliktir ve bir kenar eklemek bir ve yalnızca bir döngü oluşturur
6. G bağlantılıdır ve keyfi bir kenarın çıkarılması bağlantıyı keser Kanıtlar ?

Aşağıdaki tanımlar özellikle ağaçlar için geçerlidir.

Bir düğümün eksantrikliği $\varepsilon(u)$= $\max_{v\in V} d\ (u, v\ )$ herhangi bir v düğümüne olan maksimum uzaklıktır∈ V . Her bir düğümün eksantrikliği aşağıdaki grafikte gösterilmiştir



Bir G grafının yarıçapı rad (G)= $\min_{(u)\in V}$ $\varepsilon(u)$, V'deki tüm düğümlerin minimum dış merkezliliğidir.

Bir G grafiğinin çapı $diam$(G)= max$_{(u)\in V}$ $\varepsilon(u)$ V'deki tüm düğümlerin maksimum dış merkezliliğidir. Aynı zamanda V'deki herhangi iki düğüm arasındaki maksimum uzaklıktır.

Bir G grafının merkezi, V'deki düğümlerin minimum olduğu kümedir.

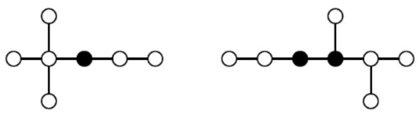
eksantriklik (siyah düğüm)

Bir T ağacının yaprağı 1.dereceden bir düğümdür

Önerme T bir ağaç olsun ve $T^{\prime}$ tüm yaprakları çıkarılarak elde edilen ağaç , o zaman $\varepsilon(T^{(\prime)})$= $\varepsilon(T$)− $T^{(\prime)}$'nin tüm düğümleri için 1.

Kanıt mı?

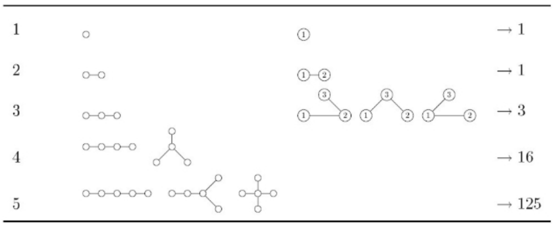
Önerme Bir ağacın merkezi tek bir düğüm veya bir çift bitişik düğümdür.



Kanıt Önceki önermeyi kullanarak tümevarım yoluyla.
Merkezin değişmediğini gösterin.

Ağaçları saymak

n düğümlü kaç farklı (etiketli) ağaç vardır? Aşağıdaki tablo küçük *n* için sayıyı vermektedir



Cayley'in aşağıdaki teoremi tam formülü verir.

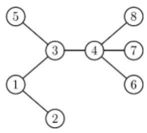
Önerme

Sırası *n* olan farklı etiketli ağaçların sayısı $n^{(n)}$'e eşittir^{-2}

Algoritma aracılığıyla bir dizi ile T_n'in bir bieksiyonunu oluşturuyoruz

Algoritma PrüferSequence(T)
s := (); *t* := ();
while E | | >1 **do**
  en küçük *i* indeksli yaprağı seçin;
  $T := T \setminus \{i\}$; $s := (s, i)$; $t := (t, komşu(i))$;
end while

Aşağıdaki grafikte, yanındaki tabloyu verir



i	s_i	t_i
1	2	1
2	1	3
3	5	3
4	3	4
5	6	4
6	7	4

Biri, grafiğin {1, . . . 'den *n*– 2 sayı olan t_i dizisinden yeniden yapılandırılabileceğini gösterir. *n}* arasında tam olarak $^{n(n)-2}$ tane böyle dizi olduğunu gösterir.

Yayılan ağaç

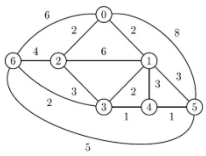
Bağlı bir grafikten bağlı kalırken mümkün olduğunca çok kenarı çıkarın; bu *n*–1 kenarlı bir ağaç vermelidir.

Bu, zaman karmaşıklığı $0(m \log m)$ olan aşağıdaki algoritma ile çözülen minimal yayılan ağaç problemidir

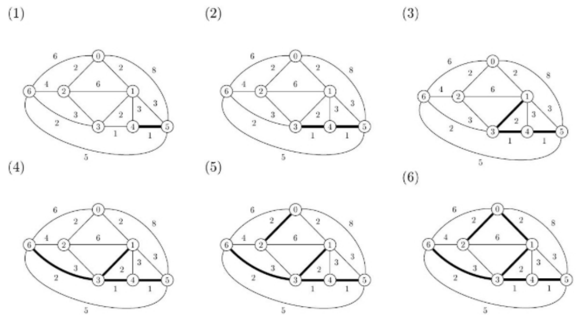
Algoritma KruskalMST(G)
$E_{ord} := sort(E)$; $E' := \varnothing$; $E_{rest} := E_{ord}$;
while $|E(')| > n-1$ **do**
  $\alpha := first(E_{rest})$; $E_{rest} := E_{(rest)} \setminus \{\alpha\}$;
  eğer $(V, E(') \cup \{\alpha\})$ asiklik **ise o zaman**
    $E' := E' \cup \{\alpha\}$;
  end if end
while

Sıralama da $0(m \log m)$ zamanında verimli bir şekilde yapılır.

Bir örneğe bakalım



Algoritmanın farklı adımları şunlardır



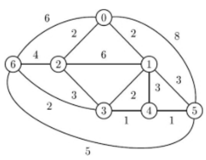
Bu, *n*– 1 kenarlı bir alt grafik olan bir ağaç oluşturur.

Şimdi zaman karmaşıklığına sahip alternatif bir algoritmaya bakalım $O((m+n) \log n)$

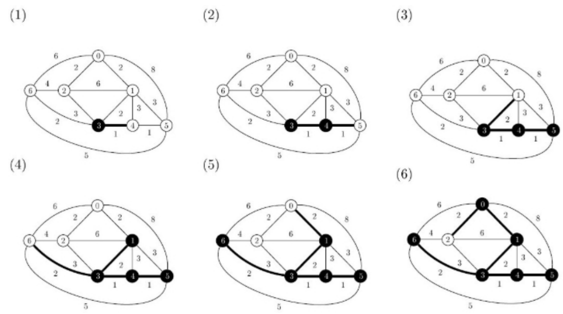
Buradaki fikir, rastgele bir düğüm seçmek ve daha sonra buradan minimal bir ağaç büyütmektir

Algoritma PrimMST(G)
$u \in V$ seçin; $V' := \{u\}$; $E' := ;\varnothing$
i **için**= 1 : n **için**– 1 **do**
  $E('') := V'$yi V''ye bağlayan kenarlar;
  $e = (u, v) \in E('')$'yi minimum ağırlıkta seçer ve öyle ki
  $(V' \cup \{v\}, E'\{e\}$ asikliktir;
  $V' := V(') \cup \{v\}$; $E(') := E' \cup \{e\}$;
için son

Şimdi aynı örneğe bakalım



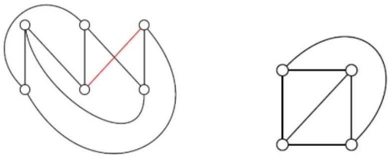
Algoritmanın farklı adımları şunlardır



(V, E') grafiği, *n*– 1 kenarlı bir minimal yayılan ağaçtır

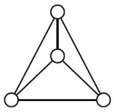
Düzlemsel grafikler

Bağlantılı graflar çizilirken doğal olarak kesişen kenarlar sorusu ortaya çıkar. Bir çizge, kesişen kenarlar olmadan çizilebiliyorsa (veya temsil edilebiliyorsa) düzlemseldir

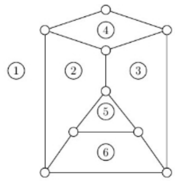


Yukarıdaki grafikler $K_{3,3}$ düzlemsel değil) ve $K_{(4)}$'ü (düzlemsel) temsil etmektedir

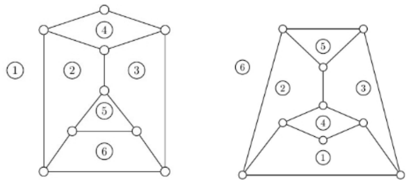
Önerme (Fary, 1948) Her düzlemsel grafik, düzlemde yalnızca düz kenarlar kullanılarak temsil edilebilir



Bu tür grafikler için artık yüzler tanımlanabilir. Bunlar, bir döngü oluşturan kenarlar tarafından çevrelenen bölgelerdir. Bu şekilde gösterildiği gibi 6 yüzlü bir dış yüz de tanımlanmalıdır

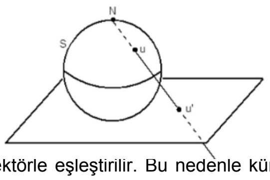


Önerme Bir grafiğin düzlemsel bir temsili, herhangi bir yüzün dış yüz haline geldiği başka bir grafiğe dönüştürülebilir (kanıt daha sonra gelecektir)

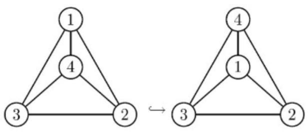


Önerme Bir grafik ancak ve ancak bir küre üzerinde temsil edilebiliyorsa bir düzlemde temsil edilebilir (daldırma)

Kanıt
Stereografik bir projeksiyon kullanın



Düzlemin her yüzü küre üzerindeki bir sektörle eşleştirilir. Bu nedenle küre üzerindeki hiçbir nokta iki farklı sektöre ait olamaz. Dış yüz, kuzey içeren bir sektörle eşleştirilmiştir.

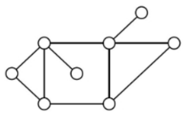


Dış yüz sonucu için, küreyi döndürerek herhangi bir noktayı (ve dolayısıyla sektörü) kuzey kutbuna taşıyabileceğimize dikkat edin

Karakterizasyon

Önerme (Euler formülü) G düzlemsel olsun ve $n(G)$ köşe sayısı, $e(G)$ kenar sayısı ve $f(G)$ yüz sayısı olsun. O zaman f $e= - n + 2$.

Burada gösterilen örnekte
$n= 8$; $e= 10$; $f= 4$



Kanıt f yüzlerinin sayısı üzerinde tümevarım yöntemini kullanın.
$f = 1$ için hiçbir döngü yoktur ve dolayısıyla bağlantılı grafik üçtür, bunun için $e= n- 1$ ve dolayısıyla $f= e- n+ 2$ olduğunu biliyoruz.
$f\geq 2$ için, $G^{\prime} := G \setminus (u, v)$ oluşturmak üzere iki yüz arasındaki bir kenarı (u, v kaldırın. Sonra $f(G^{(\prime})) = f(G)- 1$; $e(G(^{\prime})) = e(G)- 1$ ve $n(G(^{\prime}))= n(G)$. Küçük f için sonucu f için kanıtlamak için kullanın.

Bazı alıştırmalar

Önerme G, $f> 1$ ile düzlemsel olsun, o zaman $3f\leq 2e$

Önerme G, $f> 1$ ile düzlemsel olsun ve G'de hiç üçgen olmasın, o zaman $2f\leq e$

Önerme G düzlemsel (bağlantılı) bir çizge olsun. Eğer $n\geq 3$ ise $e\leq 3n- 6$

Önerme G düzlemsel (bağlantılı) bir çizge olsun. Eğer G'de hiç üçgen yoksa veya iki parçalı ise, $e\leq 2n- 4$

Bunlar aşağıdaki lemmanın kanıtlanmasına yardımcı olur

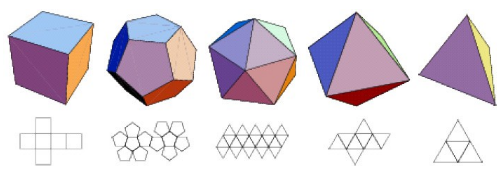
Önerme K_5 ve $K_{(3)(,)3}$ düzlemsel değildir.

Sonuç Düzlemsel bağlantılı bir G grafının köşelerinin ortalama derecesi 6'dan küçüktür $-\frac{12}{n}$

Corollary Düzlemsel (bağlantılı) bir grafikte her zaman öyle bir tepe noktası vardır ki $d(v)\leq 5$

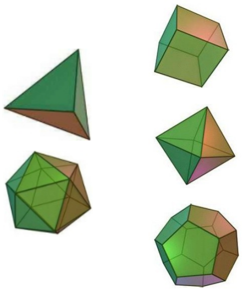
Sonuç Düzlemsel bir grafik 6 renk ile renklendirilebilir (daha sonra bakınız)

Önerme (Platonik katı) Sadece 5 düzgün çokyüzlü vardır Platonik katılar olarak adlandırılan bu çokyüzlüler aşağıda gösterilmiştir



Platonik katılar üç denklem ile karakterize edilir
k, l tam sayıları için $nk = 2e$, $fl = 2e$ ve $n + f = e + 2$
Nedenini açıklayınız
Bu durumda $2e/k + 2e/l - e = 2$ dolayısıyla $2/k + 2/l > 1$ veya $(k-2)(l-2) < 4$ olur. Tam sayı çözümleri şu şekilde verilir

İsim	*k*	*l*	*e*	*n*	*f*
Tetrahedron	3	3	6	4	4
Küp	3	4	12	8	6
Dodecahedron	3	5	30	20	12
Oktahedron	4	3	12	6	8
Icosahedron	5	3	30	12	20

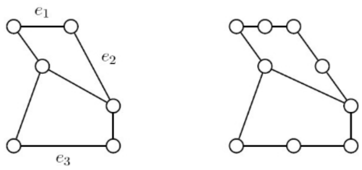


Düzlemsel grafikler için test

Öncelikle alt bölümleri ve alt grafları tanıtmamız gerekiyor.

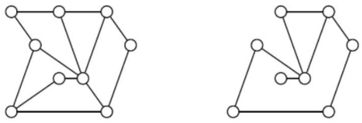
Bir $G = (V, E)$ grafını $e = (u, v) \in E$ kenarlarından birini alt bölümlere ayırarak genişletelim. E üzerine yeni bir w düğümü koyarız ve bunu iki yeni $e_1 = (u, w)$ ve $e_2 = (w, v)$ kenarı ile değiştiririz. Böylece yeni $G^{(\prime)} = (V \cup \{w\}, E \cup \{e_1, e_2\} \setminus \{e\})$ ile verilir.

Biri diğerinden bir alt bölümler dizisi aracılığıyla türetilebiliyorsa, iki grafın birbirine homeomorfik olduğu söylenir.



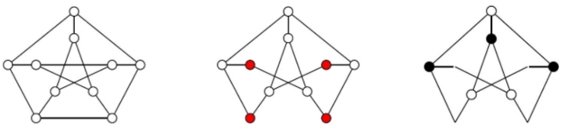
Sonuç Homeomorfizm bir denklik bağıntısıdır.

Bir $G^{(\prime)} = (V^{(\prime)}, E^{\prime})$ grafı, aşağıdaki durumlarda $G = (V, E)$ grafının bir alt grafıdır
$V^{\prime} \subseteq V$ ve $E^{\prime} \subseteq E$ (kenarlar düğümlerle birlikte kaybolmalıdır)



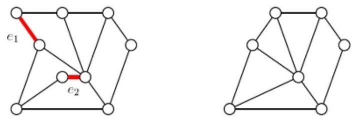
Önerme (Kuratowsky, 1930) Bir çizge, $K_{(3)(,)3}$ veya K_5'e homeomorfik bir alt çizge içermiyorsa düzlemseldir.

Örnek : Petersen grafiği (alt graf+ homeomorfizm)



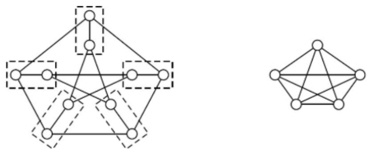
Küçükler

e olsun = (u v bir G grafiğinin bir kenarı olsun = (V E . Bir daralma
e kenarı, e'nin elenmesi ve u ve u düğümlerinin birleştirilmesinden oluşur.
v 'yi yeni bir w düğümüne dönüştürür. Yeni $G^{\prime}$ grafiği şu şekildedir
$G^{\prime} = (V \setminus \{u, v\} \cup \{w\}, E \setminus \{e\})$



Önerme (Wagner, 1937) Bir çizge, aşağıdaki özelliklere sahip değilse düzlemseldir
$K_{3,3}$ veya K_5 reşit olmayan olarak.

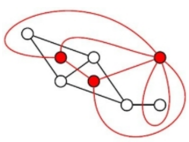
Örnek :
Petersen grafiği
tekrar



Önerme (Robertson-Seymour)
Bir G grafiği için, verilen bir H grafiğinin H'nin bir minörü olmadığını belirlemek polinom zamanda çözülebilir ($n(G)$ ve $m(G)$'ye göre).

Düzlemsel bir grafın ikili grafı G^* aşağıdaki gibi elde edilir
1. G^* G'nin her yüzünde bir tepe noktasına sahiptir
2. G'nin ilgili yüzleri arasında bir kenar varsa, G^* iki köşe arasında bir kenara sahiptir

Bu yine düzlemsel bir grafiktir ancak çoklu bir grafik de olabilir (iki köşe arasında birden fazla kenar olan)

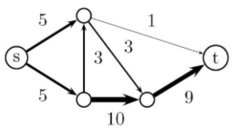


Alıştırma Euler formülünün korunduğunu gösterin

Alıştırma $G = (G^*)$ olduğunu gösterin*

Ağlar ve akışlar

Bir ağ, kaynak düğümü s olan yönlendirilmiş bir $N = (V, E)$ grafiğidir. ($d_{out}(s) > 0$ ile) ve bir terminal düğümü t ($d_{in}(t) > 0$ ile). Ayrıca her kenarın $c(e) > 0$ kesin pozitif kapasitesi vardır.

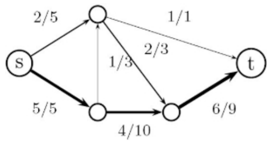


Bir f akışı : $V^2 \to \mathbb{R}^+$ her bir $e = (u, v)$ kenarı ile ilişkilidir.
1. her $e \in E$ kenarı için $0 \leq f(e) \leq c(e)$
2. her ara düğüm v için $\in V \setminus \{s, t\}$ o düğümdeki giriş ve çıkış akışı $(\Sigma_{)(u)\in(V)-((v)())} f(u, v) = \Sigma_{(u)\in(V)+((v)())} f(v, u)$ eşleşmesi

Bu durumda ağın toplam akışı F, s'den ayrılan veya t'ye ulaşan şeydir F

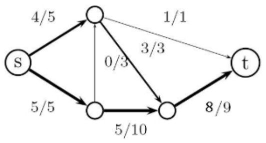
$$(N) := (\Sigma)_{u \in V} f(s, u) - (\Sigma)_{u \in V} f(u, s) = (\Sigma)_{u \in V} f(u, t) - (\Sigma)_{u \in V} f(t, u)$$

İşte bir akış örneği



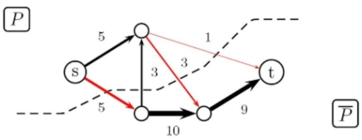
F (N)= 7 değerine sahiptir ve korunum yasası içeride doğrulanır.

Ancak akış maksimal değildir, bir sonraki ise daha sonra göstereceğimiz gibi (F(N) = 9)'dur. Bir kenarın kullanılmadığına dikkat edin (f= 0)



Bir ağın kesilmesi

Bir ağın kesimi, $V= P \cup \overline{P}$ *tepe* kümesinin P (s içeren) ve $\overline{P}$ (t içeren) olmak üzere iki ayrık kümeye bölünmesidir



Bir kesimin kapasitesi, kenarların kapasitelerinin toplamıdır (u, v) P ve $\overline{P}$ arasında

$$\kappa(P, \overline{P}) = \sum_{u \in P;\, v \in \overline{P}} c(u, v)$$

Bu da yukarıdaki örnekte 5+ 3+ 1= 9'a eşittir. Şimdi bu kapasitenin önemli özelliklerini türetiyoruz.

Önerme ($P, \overline{P}$) bir N ağının herhangi bir kesimi olsun= (V, E) o zaman ilgili akış şu şekilde verilir

$$F(N) = \sum_{u \in P;\, v \in \overline{P}} f(u, v) - \sum_{u \in P;\, v \in \overline{P}} f(v, u)$$

Kanıt Önce $F(N) = \sum_{(u) \in P} \left(\sum_{(v)} f(u, v) - \sum_{(v)} f(v, u) \right)$ olduğunu gösterin

P'deki tüm katkıları toplayarak ve korunumu kullanarak.

Tüm $v \in P$ için parantezler arasındaki terim sıfırdır (korunum). Bu nedenle sadece bölüm boyunca kenarları tutmamız gerekir.

Sonuç Bir akış, herhangi bir kesimin kapasitesi ile sınırlıdır
$F(N) \leq \kappa(P, \overline{P})$

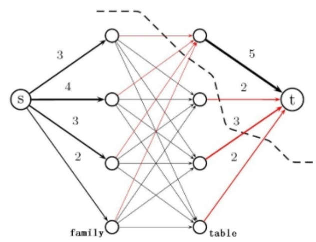
Minimal bir kesim (minimum kapasite ile) F (N)'yi de sınırlar

(bir tane oluşturacağız ve bunun aslında F(N)'ye eşit olduğunu göreceğiz)

Uygulamalar

Yemek sorunu

Üye sayısı (3,4,3,2) olan 4 aileyi koltuk sayısı (5,2,3,2) olan 4 masaya aynı ailenin iki üyesi aynı masaya oturmayacak şekilde oturtabilir miyiz?

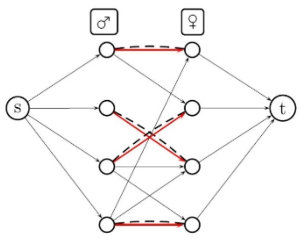


Merkezi kenarlar tablo atamalarıdır (kapasite 1). Gösterilen kesim, F (N)'yi üst sınırlayan 11 kapasiteye sahiptir. Bu nedenle dört ailenin 12 üyesinin tamamını oturtamayız.

Evlilik sorunu

Kadın ve erkekler arasında, her çiftin bir çift olup olmadığını ifade ettiği maksimum sayıda eşleşme bulmak istenir.

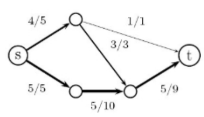
bu kuplajın kabul edilebilir olup olmadığı (var olan veya olmayan merkezi kenarlar)

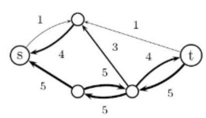


Bu yönlendirilmiş grafikte maksimum sayıda ayrık yol bulunması istenmektedir. Mevcut kenarların tüm kapasiteleri 1'dir.

Bir N(V, E) ağı ve bir f akışı verildiğinde, bunun artık ağı N_f aynı V düğümlerine sahip ancak yeni kapasitelere sahip bir ağdır

$$c_f(u, v) = \begin{cases} c(u, v) - f(u, v) & \text{eğer } (u, v) \in E\,; \\ f(v, u) & \text{eğer } (v, u) \in E \\ 0 & \text{Aksi halde.} \end{cases}$$





Bir artırma yolu, S'den $v_0, \ldots, v_k$ S'den= v_0'a
$t= v_k$ için
$\Delta_i = c(v_i, v_{(i)(+1)}) - f(v_{(i)}, v_{(i)(+1)}) > 0 \; \forall \; (v_i, v_{(i)+1}) \in E$ veya
$\Delta_i = c(v_{(i)}, v_{(i)(+1)}) - f(v_{(i)(+1)}, v_{(i)}) > 0 \; \forall \; (v_{(i)(+1)}, v_i) \in E$
Orijinal akış artırılabileceğinden bu yol optimal değildir.

Maksimum akış Min-kesim

Önerme
Akış, s'den t'ye hiçbir artırma yolu yoksa optimaldir.

Kanıt s'den u'ya bir artırma yolu varsa $u \in P$ ve aksi takdirde $u \in \overline{P}$ olan bir kesim $(P, \overline{P})$ oluşturun.
$(P, \overline{P})$'nin $F(N) = \kappa(P, \overline{P})$ için geçerli bir kesim olduğunu gösterin.

Önerme Bir N ağında aşağıdakiler eşdeğerdir

1. Bir akış en uygunudur
2. Artık grafik bir artırma yolu içermez
3. $F(N) = \kappa(P, \overline{P})$ bazı kesimler için $(P, \overline{P})$

Optimum akışın değeri böylece $F(N) = \min \kappa(P, \overline{P})$ değerine eşittir

Kanıt Okuyucuya bırakılmıştır (önceki sonuçları birleştirin)

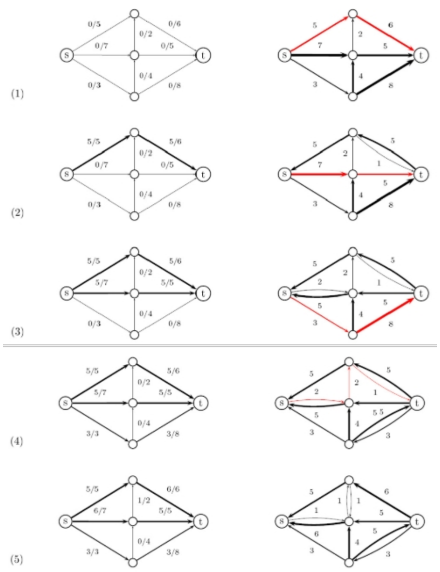
Bu, (i, j) kenarları üzerindeki x_{ij} akışlarında bir LP problemi haline gelir
max $(\sum_{(i)(:()s\, i)())} x_{si} = (\sum_{(i)(:()(i)(s\, ())} x_{is})$ $\sum_{(i} x_{ij} = (\sum_{(i} x_{ji})$ ve $0 \le x_{ij} \le c_{ij}$

Ford-Fulkerson algoritması (1956) bu optimal akışı artırma yollarını kullanarak hesaplar.

Algoritma MaxFlowFF(N,s,t)
$f(u, v) := 0 \forall (u, v) \in E$;
N_f s'den t'ye giden bir yol içerirken **do**
 s'den t'ye bir büyütme yolu A_p seçin
 $\Delta := \min_{(u,v) \in A_p} \Delta$
 A_p boyunca Δ ile akışı artırın
 $N_{(f)}$'yi güncelleyin
end while

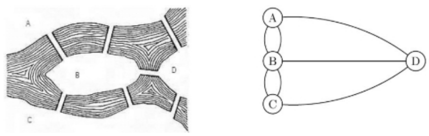
Artık grafiğinde bir yol bulmak, aşağıda gösterildiği gibi bir BFS veya DFS araştırması ile uygulanabilir

Her adımda grafiği (solda) ve artık grafiği (sağda) gösteriyoruz Büyütme yolları kırmızı renktedir. 5 adımda $F(N) = 14$ buluyoruz



Eulerian turu (1756)

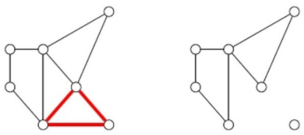
Bir Eulerian döngüsü (yolu), G'nin $G_e = (V, E_{(e)})$ alt grafiğidir $= (V, E)$
G'nin her kenarından tam olarak bir kez geçer.
Dolayısıyla G bağlantılı olmalı ve tüm V köşeleri ziyaret edilmelidir (belki de birden fazla kez). O zaman G'nin Eulerian olduğu söylenebilir



Önerme Bir G grafiği, eğer bağlıysa ve tek dereceli hiçbir köşesi yoksa bir Eulerian döngüsüne sahiptir
Bir G grafiğinin Eulerian yolu varsa (yani kapalı değilse), bağlıysa ve tek dereceli 2 veya hiç köşesi yoksa

Bu, yukarıdaki grafiğin Eulerian olmadığını kanıtlayacaktır.

Kanıt (döngülerle ilgili ilk bölümün)
Gereklilik G Eulerian olduğu için tüm düğümleri ziyaret eden bir döngü vardır. $v \in V$ her ziyaret edişimizde tekrar terk ederiz, dolayısıyla $d(v)$ çifttir. **Yeterlilik** Tek bir izole düğüm için bu önemsizdir. $|V| > 1$ için grafta bir φ döngüsü olmalıdır. Aynı düğümlere sahip ancak φ kenarları çıkarılmış H alt grafiğini düşünün.

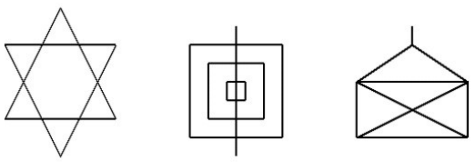


H_i bileşenlerinin her biri çift derece koşulunu sağlar ve yine bir Eulerian döngüsü φ_i'ye sahiptir. Tekrarlama yoluyla G'yi izole köşelerine indirgeyebiliriz.

Eulerian döngüsünü yeniden oluşturmak için temel bir φ döngüsünden başlayın. Başka bir φ_i döngüsünün bir düğümüyle her karşılaşıldığında, bu döngüyü düğümün yerine koyun (ve bunu özyinelemeli olarak yapın).

Kanıt (yollarla ilgili ikinci bölümün) Alıştırma olarak bırakılmıştır

Yol problemi, kaleminizi kaldırmadan bir grafik çizip çizemeyeceğinizi söyler. Bunu aşağıdaki örneklere uygulayın.



Önerme Yönlendirilmiş bir çizge $G = V, E_{(d)}$'nin bir Eulerian turu G_e vardır Eğer bağlı ve dengeli ise, yani tüm düğümleri $d_{giriş}(v) = d_{çıkış}(v)$.

Kanıt Bir alıştırma olarak bırakıldı

Fleury'nin (1883) aşağıdaki algoritması, eğer varsa bir C döngüsünü yeniden yapılandırır. $E^{\prime}$ algoritma tarafından daha önce ziyaret edilen kenarların kümesidir.

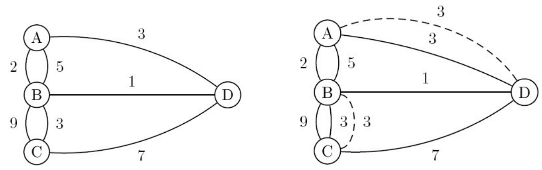
Algoritma FindEulerianCycle(G) $v_0 \in V$
seçin; $E^{\prime} := \varnothing$; $C := \langle\,\rangle$; i **için=** $1 : m$
do
  $e = (v^{(}{}_{i)-1}, v_i)$ s.t. $G^{(\prime)} = (V, E \backslash E^{(\prime)})$'nin 1 bağlantısı vardır;
  $E^{\prime} := E^{\prime} \cup \{e\}$; $C := \langle C, e \rangle$; $v_{(i)} = v_{i-1}$;
için son

Egzersiz
Eulerian Yol Problemine yönelik bir değişiklik önerisi

Peki ya grafik Eulerian değilse? Problemin minimum maliyetli bir modifikasyonunu bulabilir miyiz?

Çinli Postacı (1962)

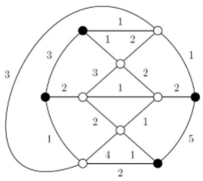
Eulerian döngü probleminin minimum maliyetli bir modifikasyonunu ele alıyoruz. Çinli bir postacının bir grafın tüm kenarlarından geçen bir tur bulması ve yolun uzunluğunu en aza indirmesi gerekir.

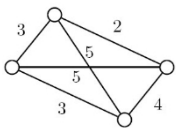


Kenarların bir maliyeti vardır ve grafiği Eulerian yapmamız gerekir

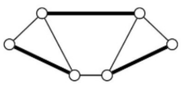
Egzersiz
1. Basit bir alt sınır verin. 2. Bu sınır ne zaman karşılanabilir?
3. Başka bir çözüm (veya daha iyi çözüm) var mı?

Çözüm : Tüm tek dereceli köşeleri bulun ve aralarındaki en kısa yolları bulun





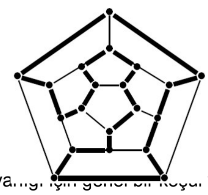
Şimdi bu grafikteki düğümlerin mükemmel bir eşleşmesini bulun.
Bir grafikte mükemmel bir eşleşme, tüm köşelerin kesiştiği bir grafiğin ayrık kenarları kümesidir.



Bu, Macar algoritması ile $0(n^3)$ zamanda çözülebilir.

Hamilton döngüsü (1859)

Hamilton tarafından 1859 yılında Dublin'deki bir oyuncak üreticisine satılan bir oyundu. Bir Hamilton döngüsü, $G_h = (V, E_h)$'nin döngüsel bir alt grafiğidir.
Tüm düğümlerden tam olarak bir kez geçen $G = (V, E)$

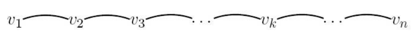


Bu zor bir problemdir ve va... için genel bir koşul yoktur (Eulerian yol probleminin aksine).
Platonik katılar ve tam grafikler için mevcuttur, ancak Petersen grafiği için mevcut değildir

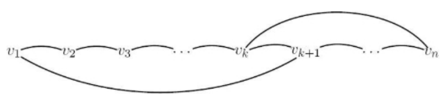
Önerme (Dirac, 1951) $n \geq 3$ düğümlü bir G grafı ve
$d(v) \geq n/2$, $v \forall \in V$, Hamiltonyen'dir

Kanıt
G bağlantılıdır, aksi takdirde en küçük bileşeni $d(v) < n/2$ olan tüm kenarlara sahip olacaktır

Daha sonra en uzun $v_1 v_{(2)} \ldots v_n$ yolunu düşünün (belki $n < |V)|$



$d(v_{(1)})$, $d(v_{(n)}) \geq n/2$ olduğundan, aynı zamanda bir döngü tarafından da kapsanmalıdır (çünkü v_1 ve $v_{(n)}$nin tüm komşuları bu yol üzerindedir)

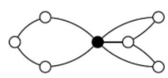


Bağlantılılık nedeniyle $n = |V|$ ve bu bir Hamilton döngüsüdür.

Alıştırma $d(v) < n/2$ olan ve yine de bir Hamilton döngüsüne sahip bir grafik oluşturun

Önerme
Eğer $G = (V, E)$ bir Hamilton döngüsüne sahipse, o zaman $G - V^{\prime}$ en fazla $|V^{\prime}|$ $V^{\prime} \subset V$ köşelerinin herhangi bir alt kümesi için bağlı bileşenler.
Kanıt H, G'nin bir Hamiltonian altgrafı olsun, o zaman $H - V^{\prime}$
$|V^{\prime}|$ bağlı bileşenlerinden daha azdır. Ancak $G - V^{\prime}$, $H - V^{\prime}$ ile aynı köşelere sahiptir ve ek kenarları vardır.

Alıştırma Bu grafikte bir Hamilton döngüsü var mı?



Alıştırma Tam iki parçalı $K_{m,n}$ grafiğinin Hamiltonyen olduğunu kanıtlayın, eğer m n=

Gezgin Satıcılar Problemi

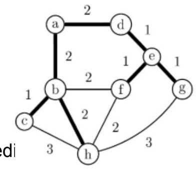
Gezgin bir satıcının bir dizi şehri (bir grafikteki düğümler) ziyaret etmesi ve seyahat süresini (veya toplam uzunluğu) en aza indirmesi beklenir

Bu NP-zordur ancak genellikle makul bir sürede yaklaşık olarak çözülebilir. $d(u, v) + d(v, w) \geq d(u, w) \forall u, v, w \in V$ üçgen eşitsizliğine sahip bir uzaklık grafiği düşünün.

Bir minimum ağırlıklı yayılan ağaç T oluşturun ve BFS kullanarak düğümleri ziyaret edin.

Bu örnek için bir döngüye sahip oluruz (a,b,c,b,h,b,a,d,e,f,e,g,e,a)

Tüm kenarların iki kez ziyaret edildiğine dikkat edi



En uygun yol P^* eşitsizlikleri karşılar *maliyet* $(T) <$ *maliyet* $(P)^* \leq 2.$*maliyet* (T)

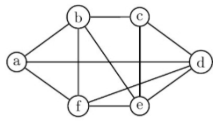
Alıştırma Nedenini açıklayın

Düzlemsel grafik için test

Bir Hamilton grafiğinin düzlemsel olup olmadığını test etmenin basit bir yolu vardır

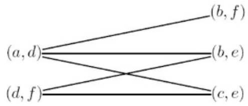
1. G'yi Hamilton grafiği H dışta olacak şekilde çizin

Aşağıdaki grafik $H = (a, b, c, d, e, f, a)$ dışta olacak şekilde çizilmiştir



2. K'yı, düğümleri $e_1, \ldots$ kenarları olan çizge olarak tanımlayın. , e_r H'de değil ve G'de kesişiyorlarsa e_i ve e_j arasında bir kenar ile.

Aşağıdaki çizge $(a, d), (b, f), b, e), c, e), (d, f)$ köşelerine ve G'deki kesişimlere karşılık gelen beş kenara sahiptir

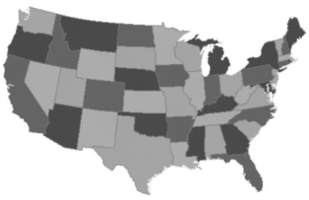


O halde G düzlemseldir, eğer K iki parçalı ise

Alıştırma Nedenini açıklayın

Dört renk sorunu

1852 yılında bir ülke haritasının (ABD haritası gibi) her zaman yalnızca dört renkle renklendirilebileceği varsayılmıştır. Nokta sınırları için altta yatan bir varsayım vardır.



Bu durum 1976 yılında K. Appel ve W. Haken tarafından kanıtlanmıştır, ancak kanıtları 1200 sözde kritik vaka üzerinde bir bilgisayar araştırması kullanmıştır.

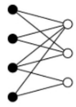
Alıştırma Altta yatan grafik hangi özelliğe sahiptir?

Renklendirme düğümleri

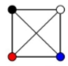
Bir $G = (V, E)$ grafının k-renklendirmesi bir $f : V \to 1, \ldots, k$ öyle ki $f(v_i) \neq f(v_j)$ eğer $(v_i, v_j) \in E$.
Bir grafiğin kromatik sayısı, bir k-renklendirmenin var olduğu en küçük k sayısıdır.

Bilinen kromatik sayılara bazı örnekler şunlardır:

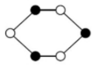
İki parçalı grafik
$\chi(G) = 2$



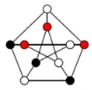
Clique
$\chi(K_n) = n$



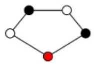
Eşit döngü
$\chi(G) = 2$



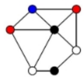
Petersen Grafiği
$\chi(G) = 3$



Tek döngü
$\chi(G) = 3$



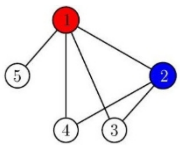
Düzlemsel grafik
$\chi(G) = 4$



Genel graflar için renklendirme problemi NP-tamdır, ancak bu tür problemler genellikle daha ilginç uygulamalara yol açar

Sınav zamanlama sorunu

Student \ Course	1	2	3	4	5
1	1	1	1		
2	1				1
3	1		1		
4	1	1		1	
5					1



Soldaki tablo her öğrencinin girdiği sınavları vermektedir İlgili grafiğin kromatik sayısı $\chi(G)$ sınavlar için minimum zaman dilimi sayısını vermektedir

Alıştırma Öğrencilerin *k adet* önceden belirlenmiş program arasından seçim yaptığı böyle bir slot problemini formüle edebilir misiniz?

Bounds

Önerme

G bağlantılı ve m E= | | olsun, o zaman $\chi(G) \leq$ 1+ 2m+ 1 2 q ____ 4

Kanıt $C = \{C_1, \ldots, C_k\}$ V'nin renklere bölümü olsun. İki renk arasında en az bir kenar vardır, bu da $m > B(k, 2)$ ve dolayısıyla $k^2 - k - 2m \leq 0$ anlamına gelir.

Önerme

$(G) = max\{d(v)\ v | \in V\}$ olsun, o zaman $\chi(G) \leq \Delta(G) + 1$ (trivial)

Önerme (Brooks, 1941)

$\chi(G) \leq \Delta(G)$ $K_{(n)}$'den farklı herhangi bir çizge veya tek bir döngü için

Önerme

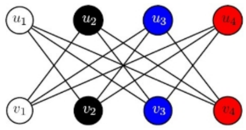
$\chi(G) \leq 1 + \max(i)\{\min(d(i), i - 1)\}$ $d_{(1)} \geq . \geq d_n$ sıralaması yapıldığında.

Kanıt Düğümleri d_i'ler gibi sıralayın ve açgözlü algoritmayı kullanın

Açgözlü algoritma

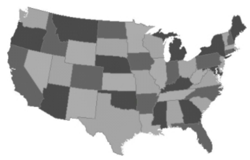
Algoritma GreedyColor(G)
L := *sort* (*V*); *c* := *sort* (*renkler*)
v **için**∈ *V* **do**
renkli komşular tarafından kullanılmayan en küçük $c_{(i)'yi}$ seçin
için son

İki parçalı bir grafikte bu açgözlü algoritma, düğümleri parça başına numaralandırırken en uygunudur, ancak $\{u(_1), v(_1), u(_2), v(_2), u(_3), v(_3), u(_4), v(_4)\}$ gibi diğer numaralandırmalar için kötü olabilir.



Egzersiz
Her grafiğin açgözlü algoritma için iyi bir numaralandırması var mı

Harita renklendirme problemine geri dönelim



ve aşağıdaki (daha basit) sonucu kanıtlamaya çalışın

Alıştırma Her düzlemsel grafik 6 renk ile renklendirilebilir

e≤ *3n*− 6 olduğunu gösterin.

O halde düzlemsel grafikler için *ortalama*(*d* (*v*)) ≤ 6− *12/n* Son olarak, *d* (*v*)≤ 5 olacak şekilde bir *v* olduğunu kanıtlayın.

Şimdi önermeyi kanıtlamak için tümevarım yöntemini kullanın (düğümleri kaldırın)

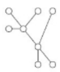
Kromatik polinom (Birkhoff-Lewis 1918)

Bir grafiğin kromatik polinomu $p_G(k)$, bir grafiğin *k* ile kaç farklı şekilde renklendirilebileceğini gösterir. Örn.

k	1	2	3	k
$p_G(k)$	0	2	24	$k(k-1)^3$

k	1	2	3	k
$p_G(k)$	0	0	6	$k(k-1)(k-2)$

k	1	2	k
$p_G(k)$	1	16	k^4



k	1	2	k
$p_G(k)$	0	2	$k(k-1)^{n-1}$

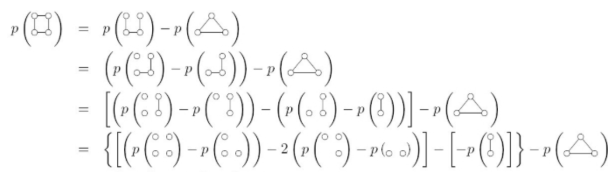
Alıştırma Yukarıdaki formülleri kanıtlayın

$\chi(G)= min\{p(G)(k)> 0\}$ olduğuna dikkat edin. Bu yardımcı olur mu?

Daha basit grafikleri kullanan güçlü bir tümevarım teoremi vardır
G− (*u, v*) (bir kenarı kaldır) ve *G*◦ (*u, v*) bir kenarı daralt)

Önerme Eğer $(u, v) \in E$ ise $p_G(k) = p_{G-((u)(,)(v)())}(k) - p_{(G)\circ(()(u)(,)(v)())}(k)$
Kanıt *u* ve *v* *G*'de farklı renklere sahiptir ve *G'*de aynıdır◦ (*u, v*)

Bu, daha karmaşık ağların kromatik polinomunu hesaplamak için kullanılabilir



$$
\begin{aligned}
&= (k^4-k^3)-2(k^3-k^2)+k(k-1)-k(k-1)(k-2) \\
&= k^4-4k^3+6k^2-3k \\
&= k(k-1)(k^2-3k+3)
\end{aligned}
$$

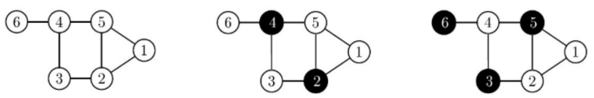
ancak sorun kombinatoryal ve dolayısıyla zor olmaya devam ediyor

Alıştırma Bir ağaç için sonucu kullanarak bunu daha hızlı türetin

Kararlı setler

Bir *G* = (*V, E*) grafında bağımsız veya kararlı bir *S* kümesi, *G'*nin herhangi bir kenarı olmayan bir alt grafıdır, yani∀ *u, v*∈ *S* : (*u, v*)∈ *E*

İki siyah düğüm kümesi sol grafiğin kararlı kümeleridir



Bu tür kümeler açıkça sadece tek bir renkle renklendirilebilir, bu da

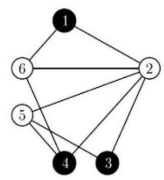
Önerme Eğer bir çizge k-kollanabilir ise, *V* şu şekilde bölümlenebilir *k* kararlı küme

Bağımsızlık sayısı $\alpha(G)$, mümkün olan en büyük kararlı kümenin boyutudur.

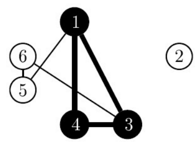
Önerme Birinde $\chi(G)$ - $\alpha(G) \geq n$ (önemsiz) vardır

Aşağıdaki örnek maksimum kararlı kümenin bulunmasını gerektirmektedir. Tabloda her bir proje için hangi öğrencilere ihtiyaç duyulduğu gösterildiğinde gerçekleştirilebilecek maksimum proje sayısını bulunuz.

Student \ Project	1	2	3	4	5	6
1	1					
2		1		1		1
3	1	1				1
4		1	1		1	
5				1	1	
6			1			
7			1			



Bunun $G_c = (V, E_{(c)})$ tamamlayıcı grafiğinde bir maksimal klik (veya tam alt grafik) bulmaya eşdeğer dikkat edin; burada E_c, *E*



Algoritma karmaşıklığı

Problemleri, onları çözmek için kullanılan algoritmalardan ayırıyoruz. Bir de zaman karmaşıklığı ve uzay karmaşıklığı meselesi var.

Bir algoritmanın $C_A(s)$ fonksiyonu, s boyutundaki bir problemi o algoritma ile çözmek için gereken zaman adımı sayısıdır.
Bir problem, aşağıdaki özelliklere sahip bir algoritma varsa polinom olarak adlandırılır
$C_A(s)$= Bazı $p(-)$ polinomları için $0(p(n))$, yani

$$\exists n_0 : C_A(s) \leq p(n) \text{ n } \quad \forall \geq n_0.$$

Farklı karmaşıklıktaki problemleri çözmek için gereken göreceli süreler

$C_A(n)$ \ n	10	20	30	40	50	60
n	0,00001 s	0,00002 s	0,00003 s	0,00004 s	0,00005 s	0,00006 s
n^2	0,0001 s	0,0004 s	0,0009 s	0,0016 s	0,0025 s	0,0036 s
n^3	0,001 s	0,008 s	0,027 s	0,064 s	0,125 s	0,216 s
n^5	0,1 s	3,2 s	24,3 s	1,7 min	5,2 min	13,0 min
2^n	0,001 s	1,0 s	17,9 min	12,7 days	35,7 years	366 cents
3^n	0,059 s	58 min	6,5 years	3855 cents	2.10^8 cents	$1,3.10^{13}$ cents

Bu, polinom problemine sahip olmanın önemini göstermektedir

Daha iyisi, makineler 100 veya 1000 kat hızlandığında çözülebilecek sorunların boyutuna bakmaktır

$C_A(n)$	size	100 times	1000 times
n	N_1	$100N_1$	$1000N_1$
n^2	N_2	$10N_2$	$31,6N_2$
n^3	N_3	$4,64N_3$	$10N_3$
n^5	N_4	$2,5N_4$	$3,98N_4$
2^n	N_5	$N_5 + 6,64$	$N_5 + 9,97$
3^n	N_6	$N_6 + 4,19$	$N_6 + 6,29$

İşte bir dizi polinom zaman problemi

İki köşe arasındaki en kısa yolu bulma Bir grafiğin düzlemsel olup olmadığını test etme
Bir grafiğin Eulerian olup olmadığını test etme Bir yayılan ağaç bulma
Mükemmel evlilik sorununu çözmek

İşte polinom olmayan bir dizi problem Bir grafiğin kromatik sayısını bulmak
Bir grafikte Hamilton döngüsü bulma Bir grafikte en büyük kararlı kümeyi bulma Gezgin satıcı problemini çözme
İki grafın izomorfik olup olmadığının test edilmesi (bilinmiyor)

Problemlerin karşılaştırılması

Bir Y problemi aşağıdaki durumlarda bir X problemine indirgenebilir (polinom zamanda)
X'in çözülmesi en az Y kadar zordur ve $X \geq_p Y$ olarak gösterilir. O halde

$X \geq_p Y$ ve $X \in P$, $Y \in P$ anlamına gelir.
$X \geq_p Y$ ve $Y \notin P$, $X \notin P$ anlamına gelir

Ağırlıkları w tamsayısı olan bir çizge üzerinde u'dan v'ye herhangi bir N uzunluğunda≥ bir yol bulma problemini [*LongestPath*(u, v, w, N)] tanımlayın

Önerme [*HamiltonianCycle*]$\leq_p$[*LongestPath*(u, v, w, N)]

Kanıt Birim ağırlık w seçin. Bir e= (u, v) kenarı seçin.
$G' = G \backslash e$ içinde N n= – 1 uzunluğunda bir en uzun yol varsa, G Hamiltonyen'dir. Tüm $m < n^2/2$ kenarları deneyin.

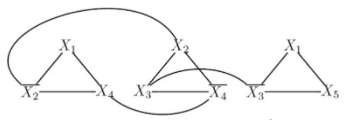
Hamiltonyen döngü probleminin P'de olmadığını bildiğimizden en uzun yol problemi de P'de değildir.

Bir Boole cümlesi, $X_i \in \{0, 1\}$ Boole terimlerinin ve bunların olumsuzlaması $\overline{X}_{(i)} \in \{0, 1\}$'in bir ayrışımıdır, örneğin $X_{(1)} \vee X_2 \vee X_4 \vee X_7$ bir 4 terimidir.
SAT] problemini, bir dizi Boole cümlesinin aynı anda sağlanıp sağlanamayacağını kontrol etmek olarak tanımlayın ([3SAT] sadece 3 terim içerir).
Örneğin $\{X_2 \vee X_{(2)}, X_2 \vee X_3 \vee X_{(4)}, X_1 \vee X_4\}$ seçilerek tatmin edilebilir.
X_1= 1, X_2 = 0, X_3 = 1, X_4 = 0.

Önerme [*SAT*]$\leq_p$[*3SAT*] ve *3SAT*]$\leq_p$[*StableSet*]

Kanıt Sadece 3 terim içeren ilk kısmı kanıtlamıyoruz.

Her 3 terim için bir üçgen oluşturun ve ardından olumsuzlukları üçgenler arasında birleştirin



Kararlı bir küme için her üçgende yalnızca bir düğüm seçebilirim. O halde $n/3$ boyutunda bir kararlı küme vardır, eğer [3SAT] doyurulabilir ise.

NP ve NP-tam

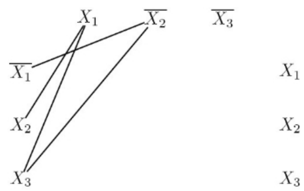
Bir çözümün geçerliliği polinom zamanda kontrol edilebiliyorsa, bir problem Deterministik Olmayan Polinom (NP)'dur.

Verilen bir döngünün Hamiltonyen olup olmadığını kontrol etmek polinom zamanda çözülebilir, ancak bunu bulmak zordur.

P'de problem polinom zamanda çözülebilir, NP'de bir çözüm polinom zamanda kontrol edilebilir.

P= NP hala açık bir sorudur (Cray ödülü = 1 milyon dolar) Bir X problemi, $X \in NP$ ve Y∀∈ NP, $Y \leq_p X$ ise NP-tamdır. **Corollary** Eğer bir NP-tam problem P'de ise o zaman $P = NP$ **Corollary** Eğer bir NP-tam problem P'de değilse o zaman $P \neq NP$

[3SAT]'ın NP-tamam olduğu bilinmektedir.
Şimdi [CLIQUE] probleminin de NP-tam olduğunu kanıtlıyoruz
[CLIQUE] problemi, G = (V, E) grafında k boyutunda klik (tam alt graf) olup olmadığını kontrol etmektir.
Kanıt $\{X_1 \vee X_2 \vee X_{(3)}, X_1 \vee X_{(2)} \vee X_{(3)}, X_{(1)} \vee X_{(2)} \vee X_3\}$. Her bir cümlenin terimlerini düğüm olarak içeren bir grafik oluşturun. Ardından, olumsuzlukları hariç tüm değişken çiftlerini bağlayın (aşağıda kısmen yapılmıştır)



Bu çizge 3 boyutlu bir klik içeriyorsa, cümle doğrulanabilirdir.

Bazı faydalı literatür

J.A. Bondy ve U.S.R. Murty, *Graph Theory with Applications*, (2. Baskı), North Holland, 1976.

Reinhard Diestel, *Graph Theory*, Graduate Texts in Mathematics, Vol. 173, Springer Verlag, Berlin, 1991.

Douglas West, *Introduction to Graph Theory*, (2. Baskı), Prentice Hall, 2000.

B. Bollobas, *Modern Çizge Kuramı*, Springer-Verlag.

Fan Cheung ve Linyuan Lu, *Karmaşık Çizgeler ve Ağlar*, Matematikte Bölgesel Konferans Serisi, Cilt 107, AMS, 2004

Dieter Jungnickel*, Graphs, Networks and* Algorithms, Algorithms and Computation in Mathematics, Vol. 5, Springer Verlag, Berlin, 2005.

25/04/2024 Ara Sınav Süre: 90 dakika İsim:

Öğrenci No:

P1 [20 puan]

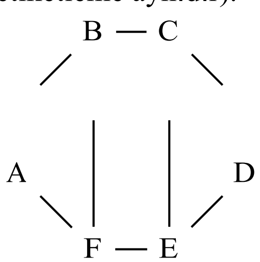
a) 6 köşesi olan 3-düzenli bir grafik çiziniz.

b) Tam iki parçalı bir grafta 13 verkesitleri için maksimum kenar sayısı nedir?

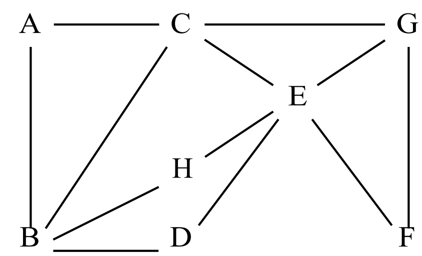
P2 [20 puan]

a) Eğer $\mathcal{K}_6$ ve $\mathcal{K}_{(5)(,)5}$ çizersek ve daha sonra $\mathcal{K}_6$'nın her köşesinden $\mathcal{K}_{(5)(,)(5)}$'in her köşesine bir kenar çizersek, son graf kaç kenara sahip olur?

b) Kaç farklı şekilde takip edilebilir? grafiği etiketlenebilir mi? İzomorfik etiketlemeler aynı kabul edilecektir. (Örneğin, grafiği yansıtırsak (y ekseni etrafında çevirirsek), soldaki D ile yeni etiketleme aynıdır).



P3 [10 puan] Aşağıdaki grafik için ölçüleri bulunuz:



G'nin eşmerkezliliği:

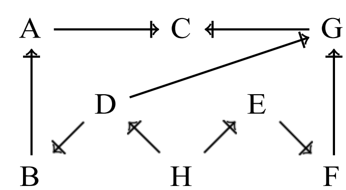
H'nin eşmerkezliliği

Yarıçap:

Çap:

Merkez:

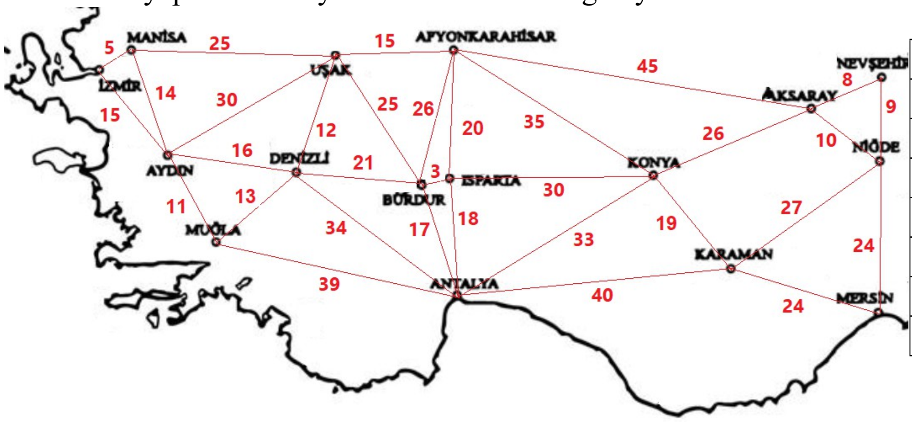
P4 [15 puan] Topolojik Sıralama ve Sayma



Grafik için topolojik bir düzen verin:

Kaç farklı topolojik düzen vardır?

P5 [15 puan] Minimum Yayılan Ağaç Aşağıdaki haritada rastgele bir şehirden başlayarak (Antalya hariç) Prim Algoritmasını kullanarak bir minimum yayılan ağaç bulun ve şehirleri MST'ye eklediğiniz sırayla yazın. Ayrıca kenarları kalın yaparak MST'yi harita üzerinde vurgulayın.



1		9	
2		10	
3		11	
4		12	
5		13	
6		14	
7		15	
8		16	

P6 [20 puan] Düzlemsel grafikler

a) Bağlantılar kesişmeden beş ev iki şebekeye bağlanabilir mi? Evet ise çizin, aksi takdirde neden olmadığını kanıtlayın.

b) Diyelim ki 8 köşeli 3-düzenli düzlemsel bir grafımız var. Bu grafın düzlemsel bir çizimi ile düzlem kaç bölgeye ayrılır? [Çizmeden yanıtlayın. Doğrudan yanıtlar 0 kredi alır, çalışmanızı gösterin].

c) Her köşe çifti için 1/2 olasılıkla bir kenar koyarak veya koymayarak rastgele 5 köşeli bir grafik oluşturursanız, düzlemsel bir elde etme olasılığı ?

d) Önce köşeleri her biri 3 köşeli iki gruba ayırarak ve sonra bir grubun her köşesinden diğer grubun her köşesine 1/2 olasılıkla bir kenar koyarak veya koymayarak rastgele 6 köşeli iki parçalı bir grafik oluşturursanız, düzlemsel bir grafik elde etme olasılığı nedir?

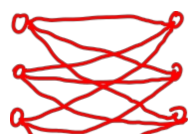
25/04/2024 Ara Sınav Süre: 90 dakika İsim:

Solutions Öğrenci No:

P1 [20 puan]

a) 6 köşesi olan 3-düzenli bir grafik çiziniz.

$K_{3,3}$ is a simple example:



b) Tam 13 vericili iki parçalı bir grafikte en fazla kenar sayısı kaçtır?

Possible such comp. bip. graphs are $K_{1,12}$, $K_{2,11}$, $K_{3,10}$, $K_{4,9}$, $K_{5,8}$ and $K_{6,7}$. Since $K_{m,n}$ has mn edges, $K_{6,7}$ would give us the maximum number of edges $6\cdot 7 = 42$.

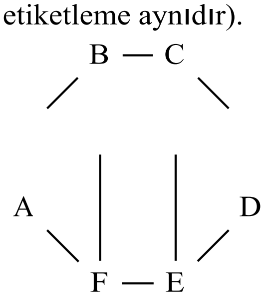
P2 [20 puan]

a) Eğer K_6 ve $K_{(5),(5)}$ çizersek ve daha sonra K_6'nın her köşesinden $K_{(5),(5)}$'in her köşesine bir kenar çizersek, nihai graf kaç kenara sahip olur?

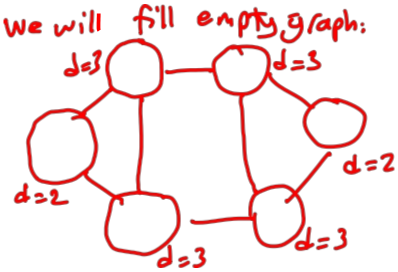
K_6 Edges in between $K_{5,5}$

$\binom{6}{2} + 6\cdot 10 + 25 = 15 + 60 + 25 = 100.$

b) Aşağıdaki kaç farklı şekilde olabilir-grafiği etiketlenebilir mi? İzomorfik etiketlemeler aynı kabul edilecektir. (Örneğin, grafiği yansıtırsak (y ekseni etrafında çevirirsek), soldaki D ile yeni etiketleme aynıdır).



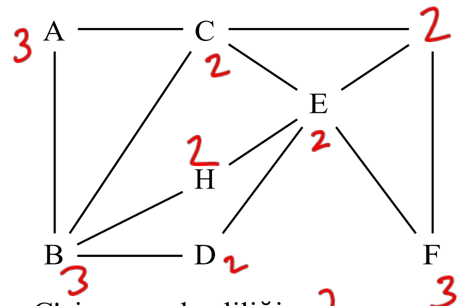
We will fill empty graph:



First choose two letters for degree 2 ones, and then place others.

$\binom{6}{2}\cdot\binom{4}{2}\cdot 2 = 15\cdot 6\cdot 2 = 180.$

P3 [10 puan] Aşağıdaki grafik için ölçüleri bulunuz:



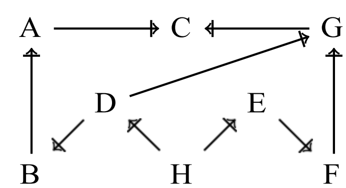
G'nin eşmerkezliliği: 2

H'nin dışmerkezliği: 2

Yarıçap 2

Çap: 3 Merkez: C, D, E, H, G

P4 [15 puan] Topolojik Sıralama ve Sayma



Grafik için topolojik bir düzen verin:

H D B E F A G → C

Kaç farklı topolojik düzen vardır?

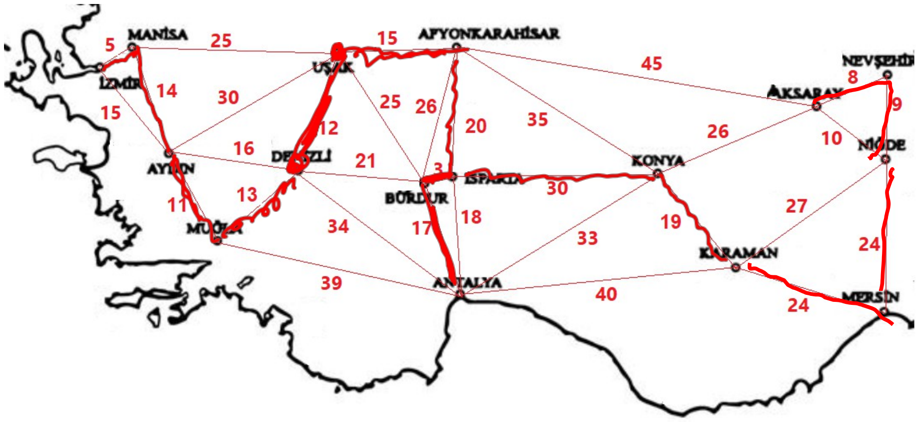
H → D → B → A → C
H → E → F → G → C

H & C's places are determined: 1st and 8th.

If D → G wasn't present, we would just choose positions of D, B, A in: H _ _ _ _ _ _ C which is $\binom{6}{3}$. (and the rest would be E, F, G)

Since we have D → G, only invalid one among these is H E F G D B A C. So, the answer is $\binom{6}{3} - 1 = 19$.

P5 [15 puan] Minimum Yayılan Ağaç Aşağıdaki haritada rastgele bir şehirden başlayarak (Antalya hariç) Prim Algoritmasını kullanarak bir minimum yayılan ağaç bulun ve şehirleri MST'ye eklediğiniz sırayla yazın. Ayrıca kenarları kalın yaparak MST'yi harita üzerinde vurgulayın.



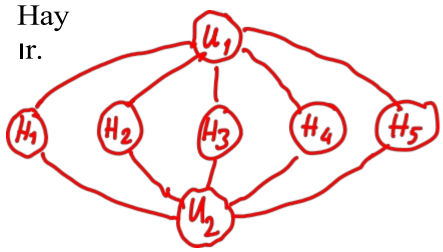
Let's choose Uşak to start:

1	Uşak	9	Burdur
2	Denizli	10	Antalya
3	Muğla	11	Konya
4	Aydın	12	Karaman
5	Manisa	13	Mersin
6	İzmir	14	Niğde
7	Afyon	15	Nevşehir
8	Isparta	16	Aksaray

P6 [20 puan] Düzlemsel grafikler

a) Bağlantılar kesişmeden beş ev iki şebekeye bağlanabilir mi? Evet ise çizin, aksi takdirde nedenini kanıtlayın Hay ır.

Yes, of course.



b) Diyelim ki 8 köşeli 3-düzenli düzlemsel bir grafımız var. Bu grafın düzlemsel bir çizimi ile düzlem kaç bölgeye ayrılır? [Çizmeden yanıtlayın. Doğrudan yanıtlar 0 kredi alır, çalışmanızı gösterin].

We have Euler's formula: $f = e - n + 2$ So, $f = 12 - 8 + 2 = 6$

3-reg: All 8 ve. has 3 outgoing edges. $3 \times 8 = 24 = 2e$, $e = 12$.

We can even check it ooo

c) Her köşe çifti için 1/2 olasılıkla bir kenar koyarak veya koymayarak rastgele 5 köşeli bir grafik oluşturursanız, düzlemsel bir grafik elde etme olasılığı nedir?

A 5-vertex graph is not planar iff it is K_5. So, to make it non-planar, we need all the possible $\binom{5}{2}$ edges. So, our probability is $1 - \frac{1}{2^{10}} = \frac{1023}{1024}$

d) Önce köşeleri her biri 3 köşeli iki gruba ayırarak ve sonra bir grubun her köşesinden diğer grubun her köşesine 1/2 olasılıkla bir kenar koyarak veya koymayarak rastgele 6 köşeli iki parçalı bir grafik oluşturursanız, düzlemsel bir grafik elde etme olasılığı nedir?

Such a graph is not planar iff all $3 \cdot 3$ edges are present giving us $K_{3,3}$. Thus, our prob. is $1 - \frac{1}{2^9} = \frac{511}{512}$

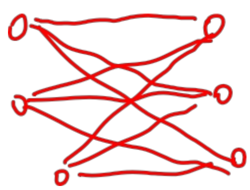
25/04/2024 Ara Sınav Süre: 90 dakika İsim:

Öğrenci No:

P1 [20 puan]

a) 6 köşesi olan 3-düzenli bir grafik çiziniz.

$K_{3,3}$



b) Tam 13 vericili iki parçalı bir grafikte en fazla kenar sayısı kaçtır?

6×7

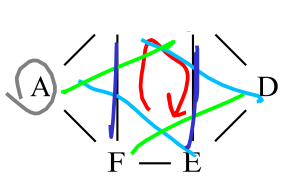
$K_{6,7} \Rightarrow 42$ edges

P2 [20 puan]

a) Eğer $\mathcal{K}_6$ ve $\mathcal{K}_{(5)(,5)}$ çizersek ve daha sonra $\mathcal{K}_6$'nın her köşesinden $\mathcal{K}_{(5)(,5)}$'in her köşesine bir kenar çizersek, son graf kaç kenara sahip olur?

$\binom{6}{2} = 15$ $6 \cdot 10 = 60$ $5 \cdot 5 = 25$ $\rightarrow 100$

b) Kaç farklı şekilde takip edilebilir? grafiği etiketlenebilir mi? İzomorfik etiketlemeler aynı kabul edilecektir. (Örneğin, grafiği yansıtırsak (y ekseni etrafında çevirirsek), solundaki D ile yeni etiketleme orijinali aynıdır).



$\binom{6}{2} \cdot \binom{4}{2} \cdot 2$

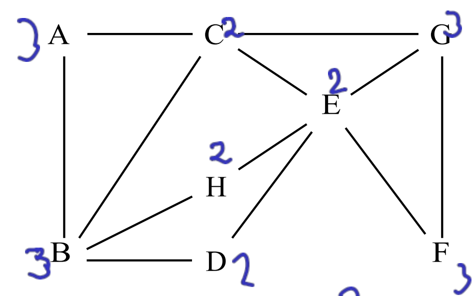
$15 \cdot 6 \cdot 2$

$\frac{4!}{2}$

$\frac{5!}{2} \cdot 3$

$60 \cdot 3 = 180$

P3 [10 puan] Aşağıdaki grafik için ölçüleri bulunuz:



G'nin dışmerkezliği 3

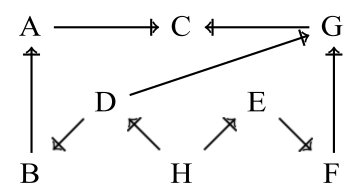
H'nin eşmerkezliliği 2

Yarıçap: Yarıçap 2 min ecc

Çap: Çap 3 max ecc

Merkez: Merkez C, E, H, D argmin

P4 [15 puan] Topolojik Sıralama ve Sayma



H E D B F A G C

5 Grafik için topolojik bir düzen verin:

H - - - - - - C

if D→G 1st

H → D → B → A → C

E → F → G

Kaç farklı topolojik düzen vardır?

$\binom{6}{3}$

$\frac{6 \cdot 5 \cdot 4}{6}$ 10

C E D

A B F

$\frac{\binom{6}{3} \binom{3}{2} \cdot 3 \cdot 2}{4}$

$\binom{6}{3} - 1$

HEFGDBAC

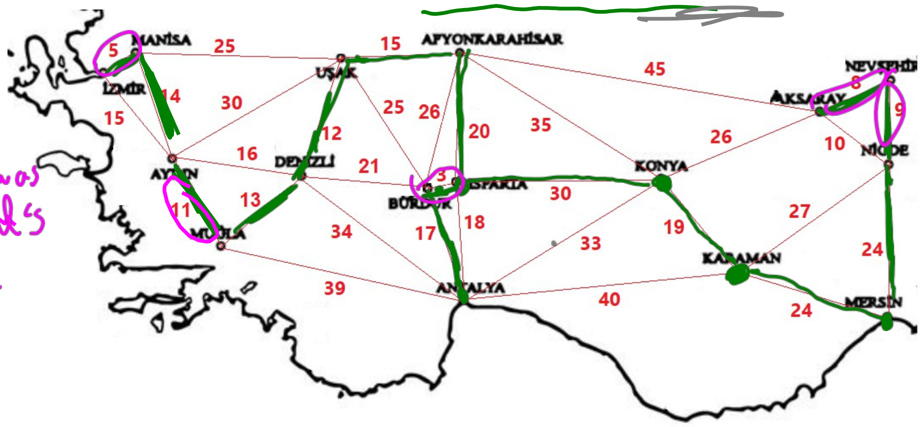
A

B — F

C — D — E

P5 [15 puan] Minimum Yayılan Ağaç Aşağıdaki haritada rastgele bir şehirden başlayarak (Antalya hariç) Prim Algoritmasını kullanarak bir minimum yayılan ağaç bulun ve şehirleri MST'ye eklediğiniz sırayla yazın. Ayrıca kenarları kalın yaparak MST'yi harita üzerinde vurgulayın.

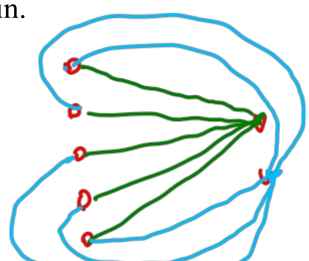
if it was Kruskal's Alg.

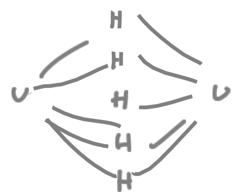


1	Karaman	9	Antalya
2	Konya	10	Afyon
3	Mersin.	11	Uşak
4	Niğde	12	Denizli
5	Nevşehir	13	Muğla
6	Aksaray	14	Aydın
7	Isparta	15	Manisa
8	Burdur	16	İzmir

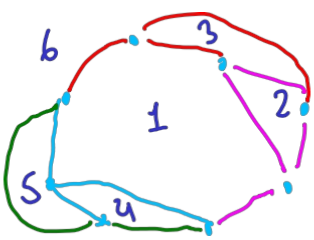
P6 [20 puan] Düzlemsel grafikler

a) Bağlantılar kesişmeden beş ev iki şebekeye bağlanabilir mi? Evet ise çizin, aksi takdirde neden olmadığını kanıtlayın.





b) Elimizde 8 köşeli 3 düzlemli bir düzlemsel grafik olduğunu varsayalım. Bu grafın düzlemsel bir çizimi ile düzlem kaç bölgeye ayrılır? [Çizmeden yanıtlayın. Doğrudan yanıtlar 0 kredi alır, çalışmanızı gösterin].



$$f = e - n + 2$$
$$= 12 - 8 + 2$$
$$= 6$$

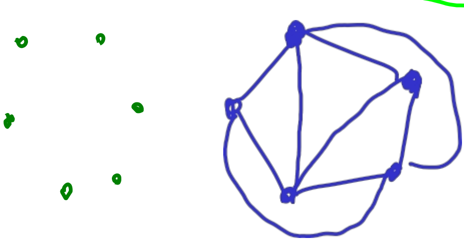
f?

$$\boxed{\Sigma d = 2e}$$

$$38 = 2 \cdot e$$

$e = 12$

c) Her köşe çifti için 1/2 olasılıkla bir kenar koyarak veya koymayarak rastgele 5 köşeli bir grafik oluşturursanız, düzlemsel bir grafik elde etme olasılığı nedir?



$$P(\text{notplanar}) = (1/2)^{\binom{5}{2}} = \left(\frac{1}{2}\right)^{10}$$

$$P(\text{planar}) = 1 - \frac{1}{2^{10}} = \frac{1023}{1024}$$

d) Önce köşeleri her biri 3 köşeli iki gruba ayırarak ve sonra bir grubun her köşesinden diğer grubun her köşesine 1/2 olasılıkla bir kenar koyarak veya koymayarak rastgele 6 köşeli iki parçalı bir grafik oluşturursanız, düzlemsel bir grafik elde etme olasılığı nedir?

$$P(\text{planar}) = 1 - \frac{1}{2^9}$$