

BLM2521 Ayırık Matematik

Discrete Mathematics

Grup-1:

Prof. Dr. Banu DİRİ

e-mail: diri@yildiz.edu.tr

<https://avesis.yildiz.edu.tr/diri>

Grup-2:

Öğr. Gör. Dr. Ahmet ELBİR

e-mail: aelbir@yildiz.edu.tr

<https://avesis.yildiz.edu.tr/aelbir>

Konular

1. Ayrık Matematiğe Giriş

2. Mantık ve İspat (Logic and Proof)

- * Önermeler (propositions)
- * Şartlı Önermeler (conditional propositions)
- * Mantıksal denklik (logical equivalence)
- * Nicelikler (quantifiers)
- * İspata giriş
- * İspat yöntemleri
- * Problem çözme

3. Boole Cebri ve Kombinatoryal Devreler (Boolean Algebras and Combinatorial Circuit)

- * Mantıksal kapılar (logic gates)
- * Boole cebri kuralları (Boolean algebra rules)
- * Karnaough haritaları (Karnaough maps)
- * Mantık devreleri (logic circuits)

4. Matematiğin Dili (The Language of Mathematics)

- * Kümeler (sets), Diziler (sequences and strings)
 - * Sayı sistemleri
-

- * İlişkiler (relations), Eşitlik bağıntıları (equivalence relations)
- * Matrisler ve fonksiyonlar

5. Sayı Teorisi (Number Theory)

- * Integer ve Asal sayılar, En büyük ortak bölen (Greatest Common Divisors)
- * Modüler aritmetik ve bölme

6. Algoritmalar, Karmaşıklık ve Asimptotik Gösterilim (Algorithms, Complexity and Asymptotic Notations)

- * Algoritma ve özellikleri
- * Karmaşıklık
- * Tekrarlı algoritmalar (iterative algorithms)
- * Rekürans Bağıntıları (recurrence relations)
- * Yinelemeli algoritmalar (recursive algorithms)
- * Farklı algoritmalar

7. Graf Teorisi (Graph Theory)

- * Yol ve Döngü (path and cycles)
 - * Hamiltonian ve Euler döngüsü
 - * En kısa yol algoritması (shortest-path algorithm)
-
- * Graflarda denkşekillilik (isomorphisms of graph)

8. Ağaçlar (Trees)

- * Terminoloji ve ağaçların özellikleri
- * Spanning tree
- * İkili Ağaç (Binary tree)
- * Tree traversals
- * Ağaçlarda denkeşlilik (isomorphisms tree)

9. Kromatik Polinomlar (Chromatic Polynomial)

- * Kurallar ve graf boyama

10. Otomata, Gramer ve Dil (Automata, Grammars and Language)

- * Sonlu durum makineleri (finite-state automata)
- * Dil ve gramer (language and grammars)
- * Nondeterministik sonlu durum makineleri

Değerlendirme

Yıl İçi : %60

Yıl Sonu: %40

Dönem Sonu Notu 40'ın altı BAŞARISIZ

Kaynaklar

- Discrete Mathematics and Its Applications Kenneth H.Rosen, McGraw Hill
- MIT Open Courseware: Mathematics for Computer Science
- İnternet kaynakları

Ayrık Matematiğe Giriş



Ayrık Matematik Nedir?

- ❑ **Ayrık matematik** (sonlu matematik), matematiğin ayrık yapılarıyla ilgilenen süreklilik içermeyen konularını kapsayan bir **matematik** dalıdır.
- ❑ Matematik sadece sürekli nesnelerle ilgilenir iken, ayrık matematik ise matematiğin ayrık nesnelerinin incelenmesine ayrılmıştır.
- ❑ Ayrık nesnelere örnek verecek olursak: tamsayılar, bir bilgisayar programı tarafından yürütülen işlem adımları, bir haritada üzerinde A noktasından B noktasına gitmek için farklı yolların tespit edilmesi verilebilir.
- ❑ Ayrık matematik dersi, bilgisayar bilimlerindeki birçok ders için gereken matematiksel altyapıyı sağlar.

Ayrık Matematik Kullanılarak Çözülen Problem Türleri

- ❑ Belirli kurallara göre bir şifre kaç yolla seçilebilir?
- ❑ Kaç tane geçerli İnternet adresi olabilir?
- ❑ Belirli bir piyangoyu kazanma olasılığı nedir?
- ❑ Bir ağdaki iki bilgisayar arasında bir bağlantı var mı?
- ❑ İstenmeyen e-posta mesajlarını nasıl tanımlayabilirim?
- ❑ Bir mesajı, istenmeyen bir alıcının okumaması için nasıl şifreleyebilirim?
- ❑ İki tamsayı toplayan bir devreyi nasıl kurabiliriz ?
- ❑ Bir ulaşım sistemi kullanarak iki şehir arasındaki en kısa yol nasıl bulunur?
- ❑ Bir grup şehirden, her birini yalnızca bir kez ziyaret eden ve turu başlangıç şehrinde bitiren en kısa turu nasıl buluruz?
- ❑ Bir bilgisayarın mantık yürütebilmesi için cümleleri nasıl temsil edebiliriz?
- ❑ Sonsuz sayıda asal sayı olduğunu nasıl kanıtlayabiliriz?
- ❑ Bir tamsayı listesi, tamsayıların artan sırada olması için nasıl sıralanabilir?
- ❑ Böyle bir sıralama yapmak için kaç adım gereklidir?
- ❑ Bir sıralama algoritmasının bir listeyi her zaman doğru sıraladığı nasıl kanıtlanabilir?

Ayrık Matematik Dersinin Amacı Nedir?

- **Matematiksel Akıl Yürütme:** Matematiksel argümanları ve ispatları okuma, anlama ve inşa etme becerisi.
- **Kombinatoryal Analiz:** Farklı türdeki nesneleri sayma teknikleri.
- **Ayrık Yapılar:** Nesneleri ve aralarındaki ilişkileri temsil eden soyut matematiksel yapılar. Örnek : kümeler, permütasyonlar, ilişkiler, grafikler, ağaçlar ve sonlu durum makineleri.
- **Algoritmik Düşünme:** Birçok problemi çözmenin tek yolu, bir algoritma belirlemektir. Algoritma, belirli bir problemin herhangi bir örneğini çözmek için izlenebilecek bir dizi adımdır. Algoritmik düşünme, algoritmaları belirlemeyi, algoritmanın yürütülmesi için gerekli olan belleği ve zamanı analiz etmeyi ve algoritmanın doğru cevabı üreteceğini doğrulamayı içerir.
- **Uygulamalar ve Modelleme:** Konuların ayrık matematikteki geniş uygulama yelpazesini anlamak ve çeşitli alanlarda yeni modeller geliştirme yeteneği önemlidir. Ayrık matematikten gelen kavramlar yalnızca hesaplamadaki problemleri ele almak için kullanılmaz, aynı zamanda kimya, biyoloji, dilbilim, coğrafya, işletme gibi birçok alandaki problemleri çözmek için de uygulanır.

Ayrık Matematik Bir Geçiş(Gateway) Dersidir...

Ayrık matematikteki konular, gelecekte alacağınız birçok ders için önemli olacaktır:

- ❑ **Bilgisayar Bilimi:** Bilgisayar Mimarisi, Veri Yapıları, Algoritmalar, Programlama Dilleri, Derleyiciler, Bilgisayar Güvenliği, Veritabanları, Yapay Zeka, Ağ Oluşturma, Grafikler, Oyun Tasarımı, Hesaplama Teorisi,
- ❑ **Matematik:** Mantık, Küme Teorisi, Olasılık, Sayı Teorisi, Soyut Cebir, Kombinatorik, Grafik Teorisi, Oyun Teorisi, Ağ Optimizasyonu, ...
- ❑ Ayrıca, öğrenilen bu kavramlar, matematiğin sürekli alanlarında da sizlere yardımcı olacaktır.
- ❑ **Diğer Disiplinler:** Burada öğrenilen kavramları felsefe, ekonomi, dilbilim ve diğer bölümlerdeki derslerde yararlı bulabilirsiniz.

Ayrık Matematik

- Ayrık matematik veya ayrık işlemsel yapılar, olası durumları sonlu olan problemin çözümü ile uğraşan matematik alanıdır.
- Bu problemler varlık belirleme, sayma ve optimizasyon olmak üzere **üç** ana kategoriye ayrılabilir.
- Bazı durumlarda çözümün var olup, olmadığı açık değildir. Bu bir varlık belirleme problemidir (existence).
- Bazı durumlarda ise çözümün olduğu bilinir ancak, bunların kaç tane olduğunu bilmek isteriz. Bu ise bir sayma (counting) problemidir.
- En iyi olan çözümün istendiği durum ise, optimizasyon problemi olarak düşünülebilir.

Kaynak: Nilüfer YURTAY, Ayrık İşlemsel Yapılar ders notları

KOMBİNASYONEL PROBLEMLER VE TEKNİKLERE GİRİŞ

Dört evli çift , her pazar akşamı iki sahada karışık ve çift olarak tenis maçı yapıyorlar. İki saat oyun süresi içinde her yarım saatte bir eşlerini ve rakiplerini değiştiriyorlar. Her bir adamın her bir kadınla en az bir kere birlikte ve karşısında oynadığı ve her bir diğer adama karşı en az bir kez oynadığı bir fikstür var mı ? Bu problem bir varlık belirleme problemidir.

Varlık Belirleme
Problemi

Altı kişilik bir yatırım kulübü her yıl başkan ve sayman pozisyonlarını dönerli olarak değiştirmek istiyorlar. Aynı insanların, aynı pozisyonlara gelmesi için kaç yıl geçmelidir ? Buradaki problem ise sayma problemi olarak düşünülebilir.

Sayma
Problemi

Bir iş yerindeki üç çalışan Ali, Ayşe ve Ahmet sırasıyla

10 milyon, 12 milyon, 15 milyon saat ücreti alıyorlar.

Patronun bu insanlara vereceği 3 ayrı iş var. Tabloda her bir insanın bu işleri ne kadar sürede yapacağı gösterilmiştir.

Patron toplam olarak en az ödeme yapacak şekilde her birine hangi işi vermelidir?

Açıkça görülmektedir ki bu bir optimizasyon problemidir.

	Ali	Ayşe	Ahmet
İş 1	7,5	6	6,5
İş 2	8	8,5	7
İş 3	5	6,5	5,5

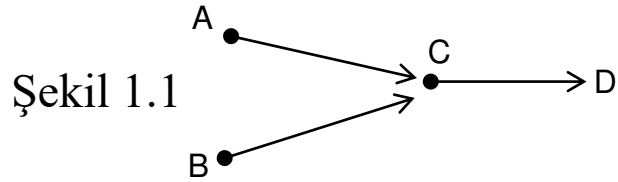
Optimizasyon
Problemi

Büyük bir mağaza bayram indirimi için 8 sayfalık bir ilan göndermeyi planlıyor. Bu ilan bayramdan en az 10 gün önce postalanması gereklidir. Ancak bir çok iş yapılması ve öncelikle de bazı kararların alınması gerekiyor. Tüm bu işlerin yapımı için zamana ihtiyaç vardır. Bunun için ise sadece 30 gün olduğunu kabul edelim. Her bir işin ne kadar süre alacağını tablo olarak verirsek;

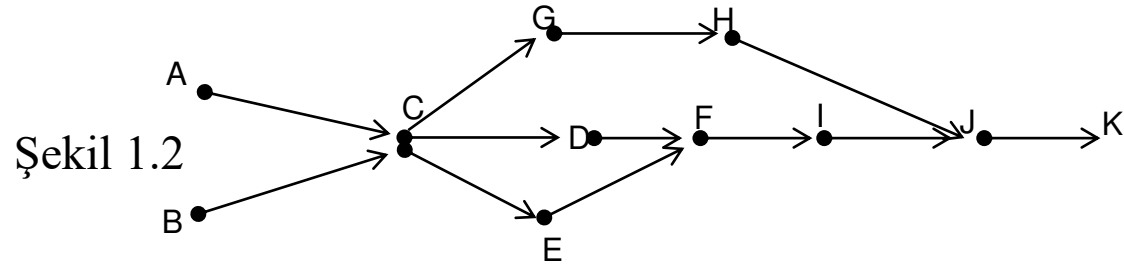
İş	Gün
Mal Seçimi (Kısım Mud.)	3
Mal Seçimi (Satın Alma)	2
İlan için mal seçimi ve fiyat belirleme	2
İlan Hazırlığı (Resim)	4
İlan Hazırlığı (Yazı)	3
İlan Tasarımı	2
Posta Listesi Hazırlama	3
Etiket Basımı	1
İlan Basımı	5
Etiket Yapıştırma	2
İlan Postalama	10

ilk bakışta bu işin
yetiştirilmesi için gerekli
toplam zaman 37 gündür.

Bazı durumlarda bilgiyi grafik olarak göstermek çok daha anlaşılır olmaktadır. Her bir görevi bir nokta ile gösterelim ve iki görev arasında eğer biri diğerinin yapılması için gerekiyorsa bir okla iki noktayı birleştirelim. Örneğin A ve B işleri C den önce yapılmalı ve D işi için de C önce yapılmalıdır. Buna göre graf şekil 1.1.'deki gibi olacaktır.



Benzer şekilde yukarıda anlatılan problemin tüm graf gösterimi şekil 1.2'deki gibidir.

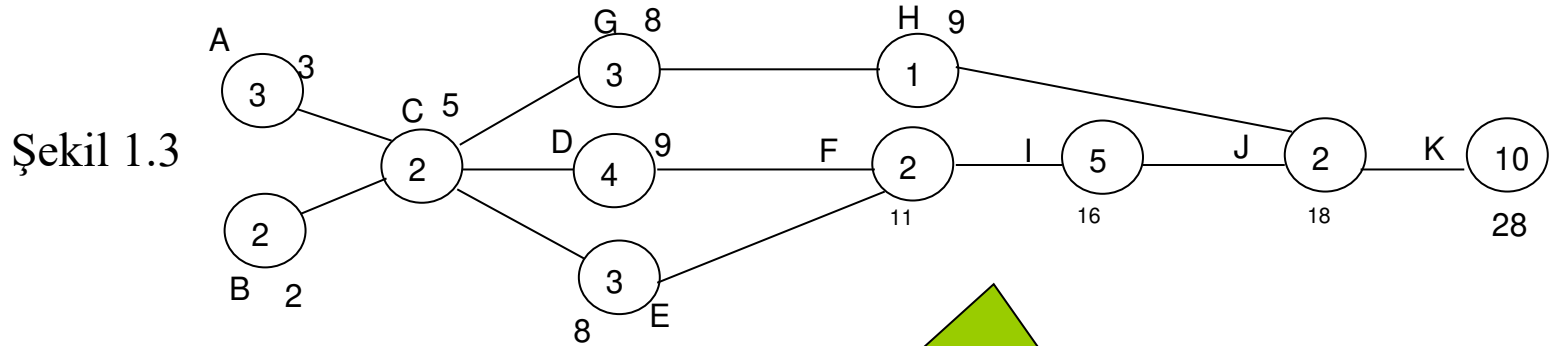


İş

Önceki İşler

A	Mal Seçimi (Kısım Mud.)	Yok
B	Mal Seçimi (Satın Alma)	Yok
C	İlana Mal Seçimi Ve Fiyat Saptama	A,B
D	Resim	C
E	Yazı	C
F	İlan Tasarımı	D,E
G	Posta Listesi	C
H	Etiket Basımı	G
I	İlan Basımı	F
J	Etiket Yapıştır	H,I
K	İlan Postalama	J

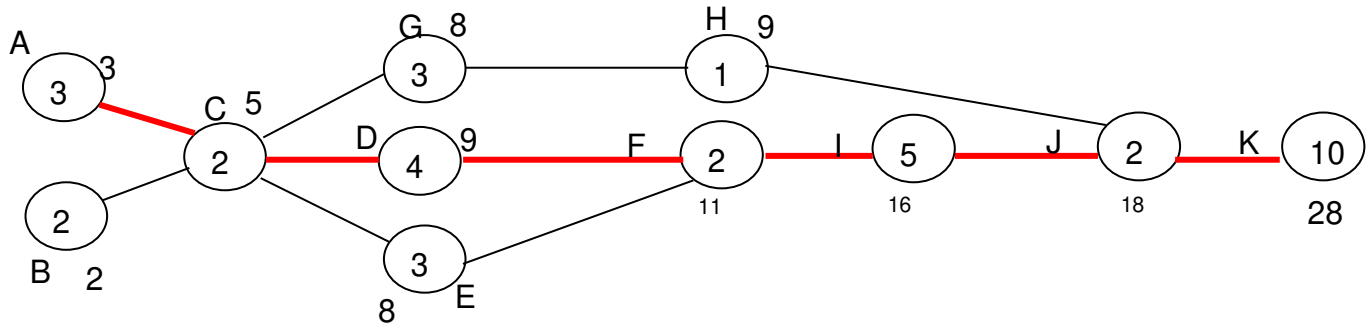
Burada tüm okların sağdan sola doğru olduğunu bildiğimize göre okları çizmeyebiliriz. Ayrıca her düğümü küçük bir daire ile gösterip içine o iş için gerekli süreyi de yazabiliriz. Bu durumda şekil 1.3'ü elde ederiz.



Sonuçta K işi için toplam 28 gün bulunduğumuza göre iş zamanında yetişecek demektir.

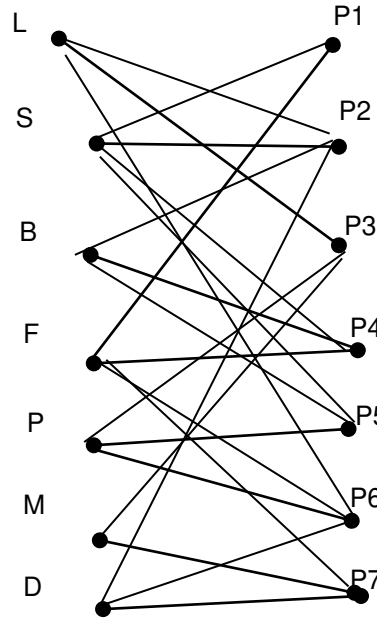
Açıkladığımız bu yöntem PERT (Program Evalution and Review Technique) olarak adlandırılır. Bu yöntem birkaç büyük projenin programlanması ve zamanlanması için çok kullanılan bir yöntemdir.

Şimdi işin tamamlandığı K düğümünden geriye doğru incelemeye başlarsak; K 28 günde tamamlanıyor, buna etken J'nin 18 günde bitmesidir. J'ye baktığımızda H ve I işlerine bağlıdır. H, 9 ve I, 16 günde bittiğine göre I çok daha önemli olmaktadır. Bu düşünce ile geriye başlangıç düğümlerine doğru analize devam edersek diyagramda sonuçta A düğümüne varırız. A-C-D-F-I-J-K yolu " Kritik Yol " olarak adlandırılır.



Türk Hava Yollarının pazartesi sabahları Ankara'dan 7 ayrı ülkeye uçuşu vardır. THY 7 ayrı pilotunu uçmak istedikleri şehirlere göre tarife yapmak istemektedir. Pilotların isteklerine göre aşağıdaki liste hazırlanmıştır:

Libya : P2 , P3 , P6
Suriye : P1 , P2 , P4 , P5
Brezilya : P2 , P4 , P5
Fransa : P1 , P4 , P6 , P7
Peru : P3 , P5 , P6
Macaristan: P3 , P7
Danimarka : P2 , P6 , P7



- ☐ Bu listeye göre görevlendirme mümkünse tüm pilotların isteklerine izin verilecek. Mümkün değilse mümkün olduğunca çok eşleme yapılacaktır.
- ☐ Bu problem bir optimizasyon problemi olarak düşünülebilir. Uçuşlarla pilotlar, pilotların isteğine göre mümkün olduğunca fazla çakışma olacak şekilde eşlenecektir.

Eşleme işlemine doğrudan başladığımızı düşünelim. Her uçuşa bir pilotu eşlemek için olası tüm yolları listeleyebiliriz ve olası her yol için kaç tane görevlendirmenin pilotların isteğine uyduğunu sayabiliriz. Örnek olarak bir alanı eşlemeyi ele alalım.

		<u>İstek</u>
L	P1	Hayır
S	P2	Evet
B	P3	Hayır
F	P4	Evet
P	P5	Evet
M	P6	Hayır
D	P7	Evet

Bu eşlemede 4 pilot istediği yere uçabilmektedir. Başka bir eşleme çok daha iyi bir sonuç verebilecektir. Eğer uçuş listesini aynı sırada sabit tutarsak, alanı eşlemede yapılan iş, pilotların sıralamasını değiştirmektir. Örneğin P2, P1, P3, P4, P5, P6, P7 bir başka eşleme olacaktır. Bu problemi çözmek için ortaya çıkan sorular şunlar :

- 1) Olası sıralama sayısı kaç tanedir ?
- 2) Tüm olası sıralamayı herhangi birini unutmadan nasıl üretebiliriz?

Kaynaklar

- “Applications of Discrete Mathematics”, John G. Michaels, Kenneth H. Rosen, McGraw-Hill International Edition, 1991.
- “Discrete Mathematic and Its Applications”, Kenneth H. Rosen, McGraw-Hill International Editions, 5th Edition, 1999.
- Nilüfer YURTAY, Ayrık İşlemsel Yapılar Ders Notları