

UFUK ÜNİVERSİTESİ · BİLGİSAYAR MÜHENDİSLİĞİ · GÜZ DÖNEMİ

MAT201

Ayrık Matematik — Discrete Mathematics

Bilgisayar Mühendisliğinin Matematiksel Mutfağına Giriş

Dr. Alaettin Uçan

Bilgisayar Mühendisi · Yapay Zeka Araştırmacısı

1. Hafta — Ders Tanıtımı ve İşleyiş Planı



Bugün Neler Konuşacağız?



Tanışma

Ben kimim, siz kimsiniz? Akademik ve sektörel arka plan.



Değerlendirme

Notlandırma, devam ve başarı kriterleri.



Vizyon

Ayrık matematik nedir, bilgisayar mühendisliği için neden hayati?



14 Haftalık Rota

Mantıktan graf teorisine dönem haritası.



İşleyiş Modeli

Quiz → Teori → Canlı Kodlama → Uygulama döngüsü.



Hazırlık Görevleri

Gelecek haftaya kadar yapmanız gereken kurulumlar.



GitHub Classroom

Ödev teslimi, otomatik testler ve anlık geri bildirim sistemi.



Soru & Cevap

Aklınıza takılan her şey.

Ben Kimim?



Dr. Alaettin Uçan

Bilgisayar Mühendisi
Yapay Zeka Araştırmacısı

E-posta: alaettin.ucan@ufuk.edu.tr

Ofis Saatleri: Pazartesi/ 16:00 – 17:00



Akademik Geçmiş

Yazılım ve Yapay Zeka Mühendisliği alanında Doktora (PhD). Ufuk Üniversitesi Bilgisayar Mühendisliği Bölümü Öğretim Üyesi.



Uzmanlık Alanları

Büyük Dil Modelleri (LLM), Doğal Dil İşleme (NLP), Edge AI (Uç Nokta Yapay Zekâsı) ve Bilgisayarlı Görü.



Sektörel Deneyim

Tiga Bilgi Teknolojileri A.Ş. — Ar-Ge Direktörü. Sağlık teknolojileri, ulusal (TÜBİTAK) ve uluslararası (Horizon Europe, Eurostars) Ar-Ge projeleri.



Bu Derse Yansıması

Üniversite–sanayi köprüsü: Derste göreceğiniz her kavramın sektörde nerede, nasıl kullanıldığını gerçek proje deneyimiyle anlatacağım.



Dersin Kimlik Kartı

DERS KODU

MAT201

Ayrık Matematik (Discrete Mathematics)

KREDİ / AKTS

3 / 6

Ulusal kredi 3, AKTS kredisi 6

DÖNEM

3. Yarıyıl

Güz dönemi · Örgün öğretim

DERSİN TÜRÜ

Zorunlu

Bilgisayar Mühendisliği lisans programı

DİL

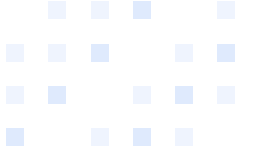
Türkçe

Terminolojinin İngilizce karşılıkları da verilir

ÖN KOŞUL

Yok

Ancak temel programlama aşinalığı işinizi kolaylaştırır



Ayrık Matematik Nedir? Sürekliden Kopuş

SÜREKLİ MATEMATİK (Analiz)

- Reel sayılar: iki değer arasında daima sonsuz değer vardır
- Türev, integral, limit — akışkan ve kesintisiz değişim
- Doğanın dili: fizik, hareket, sinyaller
- Lise ve MAT101'den tanıdığınız dünya

AYRIK MATEMATİK (Bu Ders)

- Sayılabilir, ayrık yapılar: 0 ve 1'ler, tam sayılar, sonlu kümeler
- Adım adım ilerleyen süreçler: algoritmalar, döngüler, kararlar
- Bilgisayarın dili: bellek hücreleri, ağ düğümleri, veri tabanı satırları
- Yazılımın altındaki görünmez iskelet

Bilgisayarlar süreklilik bilmez: her fotoğraf piksellere, her ses örneklere, her hesap adımlara bölünür. Bu yüzden yazılımın matematiği ayrıktır.

Bu Ders Neden Kritik? Geleceğin Derslerinin Temeli



Veri Yapıları ve Algoritmalar

Graf teorisi, ağaç yapıları ve Big-O notasyonu bu derste kurulur. VY dersinde 'linked list' gördüğünüzde arkasındaki matematik burada öğrendiğiniz bağıntılar olacak.



Kriptografi ve Siber Güvenlik

Modüler aritmetik ve sayılar teorisi, RSA gibi şifreleme sistemlerinin bel kemiğidir. İnternette her HTTPS bağlantısı bu dersin 4-6. haftalarına dayanır.



Veri Tabanı Yönetim Sistemleri

SQL'deki JOIN, aslında küme teorisindeki kesişimdir; tablolar arası ilişkiler bu derste 'bağıntılar (relations)' konusunun birebir uygulamasıdır.



Yapay Zeka ve Makine Öğrenmesi

Mantıksal çıkarım, matrisler ve graf yapıları; sinir ağlarından bilgi graflarına kadar modern yapay zekânın temel gösterim araçlarıdır.

Kural: Bu derste her soyut kavramın müfredatta bir 'müşterisi' vardır. Her konuya başlarken o müşteriyi göstereceğim.

Gerçek Dünyada Ayırık Matematik: Her Gün Kullanıyorsunuz



Google Arama

Web, sayfaların düğüm ve bağlantıların kenar olduğu dev bir graftır; PageRank bu graf üzerinde çalışır.



Sosyal Ağlar

'Tanıyor olabileceğin kişiler' önerisi, arkadaşlık grafında 2 adım uzaklıktaki düğümlerin taranmasıdır.



Navigasyon

Google Maps'in 'en hızlı rota'sı, ağırlıklı graflarda en kısa yol algoritmalarının (Dijkstra) sonucudur.



Derleyiciler

Yazdığınız if koşulunu derleyici, önermeler mantığının kurallarıyla sadeleştirir ve optimize eder.



HTTPS / E-İmza

Bankacılık uygulamanızı açtığınız anda modüler aritmetik ve asal sayılar sizi korumaya başlar (RSA).



Hata Düzeltme

QR kodların yırtık/lekeli hâlde bile okunması, sonlu cisimler üzerine kurulu kodlama teorisi sayesinde.

Dersin Öğrenme Çıktıları: Dönem Sonunda Neler Yapabileceksiniz?



- 1 Matematiksel akıl yürütme yöntemlerini (doğrudan, dolaylı, çelişkiyle ispat) açıklar ve problem çözümünde uygular.
- 2 Mantık, kümeler, fonksiyonlar ve matrislerle ilgili temel kavramları tanımlar; bu yapılarla problem formülasyonu yapar.
- 3 Sayma yöntemlerini, permütasyon-kombinasyon tekniklerini ve olasılığa temel oluşturan yapıları modellemede kullanır.
- 4 İlişkiler, graflar ve ağaç yapıları gibi ayrık kavramları tanımlar ve çeşitli sistemlerin modellenmesinde uygular.
- 5 Algoritmik düşünme becerisini geliştirir; problemleri uygun matematiksel yapılara dönüştürerek çözüm üretir.
- 6 Matematiksel kavramları farklı disiplinlerdeki problemlere aktarır ve uygun çözüm yöntemleri önerir.
- 7 Ayrık yapılar kullanarak bilgisayar bilimine ve mühendisliğe yönelik matematiksel modelleme yapar.

Bu dersin ayırt edici hedefi 5. çıktıdır: her matematiksel kavramı çalışan bir koda dönüştürebilmek.

Her Dersin 4 Aşamalı Döngüsü (2. Haftadan İtibaren)



1 • Giriş Quizi

İlk 5-10 dk

Bir önceki haftanın temel mantığını ölçen kısa, pratik bir quiz. Amaç not kırmak değil, tekrarı alışkanlık yapmak.



2 • Teorik Anlatım

~40-50 dk

Konunun matematiksel özü: tanımlar, teoremler ve bilgisayar bilimlerindeki karşılığı. Tahta + slayt.



3 • Canlı Kodlama

~30 dk

Az önce gördüğümüz formülü/yapıyı projeksiyonda birlikte, sıfırdan Python'a döküyoruz. Hatalarımız da derse dahil.



4 • Kendi Kodunuz

~30 dk

Size verilen problemi kendi bilgisayarınızda çözüyorsunuz. Ben ve asistan aranızda dolaşıyoruz: takıldığınız yerde anında destek.

Bu döngü istisnasız her hafta tekrarlanır. Derse bilgisayarınızla gelmek 3. ve 4. aşamalar için zorunludur.

Bilgisayarı olmayan öğrenciler için: laboratuvar bilgisayarları ve eşli programlama (pair programming) seçeneği mevcuttur — bana bildirin.



Laboratuvar Dilimiz: Neden Python?



Matematiğe En Yakın Sözdizimi

Küme, fonksiyon, mantık ifadeleri Python'da neredeyse matematiksel gösterimle birebir yazılır.



Sıfır Sürtünme

Derleme, proje ayarı, uzun kurulumlar yok. Fikirden çalışan koda saniyeler içinde geçilir.



Kariyer Değeri

Yapay zekâ, veri bilimi ve otomasyon dünyasının ortak dili. Bu derste yazdığınız her satır CV'nize yatırımdır.



Ön Bilgi Şartı Yok

Programlamaya yeni başlayanlar için 2. haftada hızlandırılmış bir Python başlangıç dokümanı paylaşacağım.

```
matematik_python.py
```

```
# Matematik:  $A = \{x \in \mathbb{Z} : 0 \leq x < 10, x \text{ çift}\}$   
A = {x for x in range(10) if x % 2 == 0}
```

```
# Matematik:  $A \cap B, A \cup B$   
B = {0, 3, 6, 9}  
print(A & B)    # kesişim  $\rightarrow \{0, 6\}$   
print(A | B)    # birleşim
```

```
# Matematik:  $p \wedge (q \vee r)$   
p, q, r = True, False, True  
print(p and (q or r)) #  $\rightarrow \text{True}$ 
```



GitHub Classroom: Kağıt-Kalem Ödev Devri Bitti

Tüm pratik çalışmalar ve kodlama ödevleri GitHub Classroom üzerinden yürüyecek. E-posta ile dosya göndermek veya çıktı teslim etmek yok.



Nedir?

GitHub'ın eğitim aracı: her öğrenciye, her ödev için şablondan türetilmiş özel (private) bir kod deposu açar. Kodunuzu sadece siz ve ben görürüz.



Ne Kazandırır?

Versiyon kontrolü (git) refleksi: her değişikliğin kaydı tutulur, 'kodum silindi/kayboldu' diye bir şey kalmaz. Tüm dönem çalışmalarınız portfolyonuz olur.



Teslim Nasıl İşler?

Son teslim tarihi anında depodaki son push neyse, o değerlendirilir. Sistem saati esastır; 'internetim gitti' riskine karşı son güne bırakmayın.



Adil ve Şeffaf

Herkes aynı iskelet koddan başlar, aynı testlerle değerlendirilir. Notunuzu ben değil, kodunuzun geçtiği test sayısı belirler.

Otomatik Değerlendirme (Auto-Grading) Nasıl Çalışır?



1 • Kabul Et

Ödev davet linkine tıklarsınız; GitHub Classroom size özel depoyu (iskelet kod + fonksiyon tanımları) otomatik oluşturur.



2 • Kodla

İstenen algoritmayı (örn. Öklid, doğruluk tablosu üretici) fonksiyonların içine Python ile yazarsınız.



3 • Push Et

git add → commit → push.
Kodunuz GitHub'daki deponuza yüklenir.



4 • Testler Koşar

GitHub Actions arka planda pytest'i çalıştırır; kodunuz onlarca hazır test senaryosundan geçirilir.



5 • Anlık Not

Dakikalar içinde hangi testin geçtiğini, hangisinin neden kaldığını görürsünüz. Teslim tarihine kadar sınırsız deneme hakkınız var.

En önemli avantaj: Geri bildirim döngüsü haftalar değil, dakikalar sürer. Yanlışınızı anında görür, düzeltir, yeniden denersiniz — öğrenme tam olarak bu döngüde gerçekleşir.

Örnek Ödev: Size Ne Verilir, Sizden Ne Beklenir?



SIZE VERİLEN — obek.py (iskelet kod)

```
def ebob(a: int, b: int) -> int:
    """Öklid algoritmasıyla EBOB
    hesaplar."""
    # TODO: Kodunuzu buraya yazın
    ...
```

SİZİ BEKLEYEN — test_obek.py (otomatik test)

```
def test_temel():
    assert ebob(48, 18) == 6
def test_asal():
    assert ebob(17, 5) == 1
```

Push sonrası dakikalar içinde göreceğiniz geri bildirim:



test_temel — GEÇTİ
ebob(48, 18) doğru sonuç verdi.



test_asal — GEÇTİ
Aralarında asal durum doğru.



test_negatif — KALDI
ebob(-4, 6) için beklenen 2, dönen -2. Negatif girdileri düşündünüz mü?

Not = geçen test oranı. Kırmızı bir test, kaybedilen puan değil; henüz düşünmediğiniz bir durumun size fısıldanmasıdır.



Notlandırma: Başarı Kriterleri

%40

ARA SINAV

Analitik ve teorik problem çözme. Kağıt üzerinde: tanımlar, ispat mantığı, el ile algoritma yürütme.

%60

FİNAL SINAVI

Tüm müfredatı kapsayan genel değerlendirme. Dönem boyunca döngüyü takip eden için sürpriz içermez.

%10

ÖDEVLERİN FİNALE ETKİSİ

Haftalık GitHub ödevlerinden ve quizlerden toplanan puanlar, final notunuza %10 oranında etki eder.

Devam ve katılım: Canlı kodlama ve sınıf içi uygulamalar ders esnasında yapıldığı için derse katılım kritik önemdedir. Kaçırılan bir uygulama seansı, evde videodan telafi edilebilecek bir şey değildir.



Bu Dersi Geçmenin (ve Sevmenin) Altın Kuralları



Geride Kalmayın

Ayrık matematik kümülatiftir: 3. haftayı kaçıran, 5. haftada yabancı dil dinler gibi hisseder. Takıldığınız hafta, aynı hafta yardım isteyin.



Elle ve Kodla Çözün

Konuyu 'anladım' hissi yanıltıcıdır. Bir kavramı gerçekten öğrendiğinizin kanıtı, onu hem kağıtta hem kodda çalıştırabilmenizdir.



Soru Sormak Puandır

Yanlış cevap ve 'saçma' soru diye bir şey bu sınıfta yoktur. Canlı kodlamada hata yapmak dersin tasarımının parçasıdır.



Quizleri Ciddiye Alın

Haftalık quizler küçük ama dönem sonundaki gerçek farkı yaratan şey, 14 haftaya yayılmış bu düzenli tekrardır.

Hedefim kimseyi elemek değil; dönem sonunda hepinizin 'matematik böyle işlenirmiş' demesi.

Yapay Zekâ Kullanım Politikası: Yasak Değil, Bilinçli Kullanım



Bir yapay zekâ araştırmacısı olarak ChatGPT/Claude gibi araçları yasaklamayacağım — sektörde hepimiz kullanıyoruz. Ama aradaki çizgi nettir:



ÖĞRENME ARACI OLARAK — SERBEST

- Anlamadığınız kavramı farklı örneklerle açıklamak
- Kendi yazdığınız koddaki hatayı birlikte ayıklamak
- Çözdüğünüz sorunun alternatif çözümlerini incelemek
- Kendinize ek alıştırma soruları üretmek



CEVAP MAKİNESİ OLARAK — İHLAL

- Ödev sorusunu yapıştırıp çıkan kodu göndermek
- Anlamadığınız / açıklayamadığınız kodu teslim etmek
- Quiz ve sınavlarda her türlü yapay zekâ kullanımı
- Bir arkadaşın (veya YZ'nin) çözümünü değişkenlerini değiştirerek sunmak

Turnusol testi: Teslim ettiğiniz kodu, satır satır ne yaptığını açıklayamıyorsanız o kod sizin değildir. Sözlü doğrulama yapma hakkımı saklı tutuyorum.



Akademik Dürüstlük ve İş Birliği Sınırları



Tartışmak serbest

Ödev problemini arkadaşlarınızla fikir düzeyinde tartışabilirsiniz: yaklaşım, algoritma seçimi, kavramlar. Beyaz tahta başında beyin fırtınası teşvik edilir.



Kod paylaşmak yasak

Kodun kendisini paylaşmak, göstermek, göndermek — hangi yönde olursa olsun — kopya kapsamındadır. Veren de alan da aynı yaptırıma tabidir.



Benzerlik denetimi

Tüm teslimler otomatik kod benzerlik analizinden geçer. Değişken adı değiştirme, satır sırası oynama gibi 'makyajları' bu araçlar kolayca yakalar.



Yaptırım

İhlaller Ufuk Üniversitesi öğrenci disiplin mevzuatına göre işlem görür; ilgili ödev/sınav notu iptal edilir. İlk hafta bunu netleştirmek, dönem sonunda kimseyi üzmemek içindir.

Basit ölçüt: "Yardım" bittiğinde elinizde fikir kalıyorsa iş birliğidir; kod kalıyorsa kopyadır.



Temel Referans Kaynaklarımız



ANA KAYNAK — **Kenneth H. Rosen**, *Ayrık Matematik ve Uygulamaları* (7. Baskıdan Çeviri, Palme Yayıncılık)

Dünyada bu dersin fiilî standardı. Haftalık okuma bölümleri ve alıştırmalar numaraları bu kitaptan duyurulacak; edinmenizi şiddetle öneririm.

DESTEKLEYİCİ KAYNAKLAR (kütüphane / tercihe bağlı)

- **Grimaldi, R.P.** — Ayrık ve Kombinatoriyal Matematik: Uygulamalı Bir Giriş (5. Baskı) — sayma konularında ekstra derinlik
- **Johnsonbaugh, R.** — Ayrık Matematik (7. Baskı) — bol çözümlü örnek arayanlar için
- **Ross, K.A. & Wright, C.R.B.** — Ayrık Matematik (4. Baskı) — titiz ve öz anlatım
- **Epp, S.S.** — Uygulamalı Ayrık Matematik (1. Baskı) — ispat yazmayı adım adım öğreten en iyi kaynaklardan
- **Anderson, J.A.** — Kombinatorik ile Ayrık Matematik (2. Baskı) — kombinatorik ağırlıklı alternatif

Ayrıca: Ders slaytları, canlı kodlama dosyaları ve haftalık ödev depoları — hepsi GitHub üzerinden sizinle paylaşılacak.



İlk Yarı: Temelleri Döşüyoruz (1–7. Haftalar)

1–3
HAFTA

Mantık, Kümeler ve Fonksiyonlar

Matematik: Önergeler, doğruluk tabloları, mantıksal denklıklar; küme işlemleri, fonksiyon türleri.

Kodda: Python'da bool cebiri, set/dict yapıları ve kendi doğruluk tablosu üreticinizi yazmak.

4–6
HAFTA

Algoritmalar, Tam Sayılar ve Matrisler

Matematik: Algoritma kavramı, Big-O notasyonu, modüler aritmetik, Öklid algoritması, matris işlemleri.

Kodda: Kod performansını ölçmek; RSA şifrelemenin matematiksel çekirdeğini kendi elinizle kurmak.

7
HAFTA

Matematiksel Akıl Yürütme ve İspat

Matematik: Doğrudan ispat, dolaylı ispat, çelişkiyle ispat; karşı örnek bulma stratejileri.

Kodda: assert ifadeleri ve döngü değişmezleri (loop invariants) ile 'kodun doğruluğunu ispatlamak'.

İlk yarının sonunda: mantıksal düşünen, karmaşıklığı ölçebilen ve iddiasını hem kağıtta hem kodda savunabilen bir mühendis adayı.



İkinci Yarı: Yapıları İnşa Ediyoruz (8–14. Haftalar)

8

HAFTA

Doğrulama Teknikleri ve Çıkarımlar

Matematik: Matematiksel tümevarım (induction), güçlü tümevarım, özyinelemeli tanımlar.

Kodda: Tümevarım $\leftrightarrow$ özyineleme (recursion) ikizliği: faktöriyel ve Fibonacci üzerinden.

9–10

HAFTA

Sayma (Kombinatorik)

Matematik: Permütasyon, kombinasyon, güvercin yuvası ilkesi, olasılığa giriş.

Kodda: Tüm olasılıkları kaba kuvvetle (brute-force) üreten kodlar: itertools ile sayma deneyleri.

11

HAFTA

İlişkiler (Relations)

Matematik: Bağlantı özellikleri: yansıma, simetri, geçişlilik; denklik bağıntıları.

Kodda: Warshall algoritmasıyla geçişli kapanışı hesaplamak; veri tabanı ilişkilerine köprü.

12–14

HAFTA

Graf Teorisi ve Ağaç Yapıları

Matematik: Graf türleri, yollar, düzlemsel graflar; ağaçlar ve dolaşma (traversal).

Kodda: BFS ve DFS'in matematiksel altyapısı; komşuluk matrisiyle kendi graf motorunuzu yazmak.

Dönemin finali graf teorisi: veri yapıları dersinin en zorlu konusuna, matematiksel temelini almış olarak gireceksiniz.

Gelecek Haftaya Kadar: 4 Adımlık Kurulum Görevi



1 Python 3.x Kurun

python.org adresinden güncel Python 3 sürümünü indirin. Kurulumda "Add Python to PATH" kutusunu işaretlemeyi unutmayın.

Doğrulama: `python --version`

2 Bir IDE Edinin

Tercihen VS Code (ücretsiz) veya PyCharm. VS Code kuranlar Python eklentisini de yüklesin.

code.visualstudio.com

3 GitHub Hesabı Açın

github.com üzerinden ücretsiz hesap oluşturun. Kullanıcı adınızı profesyonel seçin; bu hesap mezuniyette portfolyonuz olacak.

Öneri: `ad-soyad tabanlı kullanıcı adı`

4 Git Temellerine Bakın

git clone, git add, git commit, git push komutlarının ne yaptığına göz atın. Bilmeyenler için adım adım kurulum ve kullanım dokümanı paylaşacağım.

Doğrulama: `git --version`

Bu hazırlıklar gelecek haftaki ilk uygulamalı ders için zorunludur. Kurulumda takılan, dersten önce bana yazsın — ilk 15 dakikayı kurulum sorunlarıyla harcamayalım.



Takıldığınızda: Yardım Kanalları



Ders İçi

En değerli kanal. Canlı kodlama ve uygulama seanslarında el kaldırın; sorunuz büyük olasılıkla 10 kişinin daha sorusudur.



Ofis Saatleri

Pazartesi 16:00 – 17:00 arası ofisime randevusuz gelebilirsiniz. Ödev, konu ya da genel kariyer sohbeti — kapı açık.



E-posta

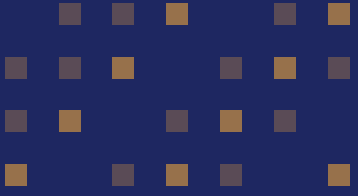
alaettin.ucan@ufuk.edu.tr— 48 saat içinde dönüş yaparım. Konu satırına mutlaka "MAT201" yazın; kod sorularında hata mesajını ve repo linkinizi ekleyin.



GitHub Issues

Ödev deponuzda 'Issue' açarak teknik soru sorabilirsiniz; cevap yazılı kalır, sınıf arkadaşlarınız da (kod paylaşmadan) faydalanır.

Tek kural: Sormadan önce en az 20 dakika kendiniz uğraşın; sorduğunuzda ise ne denediğinizi anlatın. "Çalışmıyor" bir soru değildir, "şunu denedim, şu hatayı aldım" bir mühendis sorusudur.



Sorularınız?

Şimdi sorun — ya da ofis saatinde, e-postayla, GitHub'da. Soru sormak bu dersin müfredatına dahildir.

GELECEK HAFTA

Mantık, Kümeler ve Fonksiyonlar — I

İlk canlı kodlama var: bilgisayarınız kurulu şekilde gelin.

