



DERS BİLGİ FORMU

FAKÜLTE / ENSTİTÜ ADI	Fen Edebiyat Fakültesi
BÖLÜM / PROGRAM / ANABİLİM DALI ADI	Matematik
DERSİN ADI	Bilgisayar Programlama 2
DERSİN KODU	MAT4560
YEREL KREDİSİ	3
AKTS KREDİSİ	5
HAFTALIK DERS SAATİ	3
HAFTALIK UYGULAMA SAATİ	0
HAFTALIK LABORATUVAR SAATİ	0
ÖNKOŞULLAR	Yok
YARIYIL	Güz, Bahar
DERSİN DİLİ	İngilizce, Türkçe
DERSİN SEVİYESİ	Lisans
DERSİN TÜRÜ	Seçmeli @Matematik Lisans Programı Seçmeli @ Fen Bilgisi Eğitimi Lisans Programı
DERSİN KATEGORİSİ	Uzmanlık/Alan Dersleri
DERSİN VERİLİŞ ŞEKLİ	Yüz Yüze
DERSİ SUNAN AKADEMİK BİRİM	Matematik Bölümü
DERSİN KOORDİNATÖRÜ	Sema AKIN BAŞ
ASİSTAN(LAR)	
DERSİN AMACI	Bu dersin amacı, sıfırdan programlamaya başlanarak Python dilinde temelden ileri seviyeye gelmesidir. Ders aynı zamanda, öğrencilerin verilen bir problemi anlayıp algoritmasını hazırlayabilme, hazırlanan algoritmayı Python programlama dilinde en uygun şekilde kodlayabilme, yazılan programın algoritmasında ve derleme esnasında oluşan hatalarından arındırabilme becerilerini kazandırmayı amaçlamaktadır.
DERSİN İÇERİĞİ	Ele alınan programlama dilinin (Python) yazım kuralları; Python komutları; Python'da veri yapıları, değişkenler, operatörler ve temel kontrol yapıları (if, for, while); Python'da fonksiyon; modüler programlama prensipleri; Python'da veri analizi için Pandas ve NumPy; Python ile veri analizi uygulamaları; Python'da veri görselleştirmesi için Matplotlib. Ayrıca, öğrenciler ödev verilen bazı problemlerin veya yöntemlerin algoritmalarını anlayıp, kodlayarak ödevler hazırlar ve sunarlar.
DERS KİTABI / MALZEMESİ / ÖNERİLEN KAYNAKLAR	Ders Kitabı: [1] Çobanoğlu, Bülent. <i>Herkes için Python</i> . Pusula Yayınevi, 2022. [2] Hunt , John. <i>A Beginners Guide to Python3 Programming</i> . Springer, 2020. [3] McKinney, Wes. <i>Python for Data Analysis</i> . O'Reilly Media, Inc., 2022. [4] Padmanabhan T. R. <i>Programming with Python</i> . Springer, 2016. Zorunlu Kaynaklar: [1] https://www.anaconda.com/download Önerilen Kaynaklar:



Hunt, John. *Advanced Guide to Python 3 Programming*. Springer, 2019.
Shaw, Zed A., *Learn Python the Hard Way*. Addison-Wesley Professional, 2012.

Ders Öğrenim Çıktıları

Bu dersi başarıyla tamamlayan öğrenciler,

1. Derleyici editörünü kullanabileceklerdir.
2. Programlama dilinin kurallarını dikkate alarak komutlarını uygulayabileceklerdir.
3. Bir problemin çözüm yöntemini belirleyerek kodlayabileceklerdir.
4. Hazırlanan programın hatalarını düzeltebileceklerdir.
5. Veri analizi ve veri görselleştirme uygulamaları yapabileceklerdir.
6. Veri bilimi ve makine öğrenmesine zemin hazırlayacak temel Python programlamalarını gerçekleştirebileceklerdir.

DEĞERLENDİRME SİSTEMİ

Etkinlikler	Sayı	Katkı Payı
Devam/Katılım		
Laboratuvar		
Uygulama		
Arazi Çalışması		
Derse Özgü Staj		
Kısa Sınavlar/Stüdyo Kritiği (Zorunlu): • İçerik: Sınav haftasına kadar işlenen konuların tümünü kapsayan kapsamlı soruların sorulması • Format: Yüz yüze. Çoktan seçmeli / boşluk doldurma kısa sınav (15-30 dakika) • Detaylı Değerlendirme Kriterleri: -Derste işlenen konuların ayrıntılarını fark edebilme ve uygulamasını yapabilme	3	%15
Ödev: • İçerik: Derste işlenen temel kavramların eleştirel biçimde yorumlanmasını ve ilgili kavramların disiplin-içi ve disiplinler arası alanlarda örneklerinin bulunmasını içeren ödevin verilmesi • Format: Yazılı raporlar ve grup sunumları • Detaylı Değerlendirme Kriterleri: - Bir problemin çözüm sürecini mantıklı ve doğru bir şekilde yazabilme - Kavramların uygulamadaki örneklerini bulabilme - Uygulamalı düşünebilme, yorumlama ve gerekçelendirme süreçlerinin yürütülebilmesi	1	%5
Sunum/Jüri: • İçerik: Öğrencilerin kendi öğrenme süreçlerini değerlendirmelerinin ve grup sunumları yapmalarının istenmesi • Format: Grup sunumları • Detaylı Değerlendirme Kriterleri:	1	%10



- Öğrenilen konuların doğru bir şekilde açıklanabilmesi -Sunum tekniklerinin doğru kullanılması		
Proje		
Seminer/Workshop		
Ara Sınavlar: • İçerik: Sınav haftasına kadar işlenen konuların tümünü kapsayan kapsamlı sorular • Format: Yüz yüze. Sınav (60 dakika) • Detaylı Değerlendirme Kriterleri: -Dersin temel kavramlarının anlaşılmasının gösterilmesi -Teorik konularla ilgili problemlerin çözülebilmesi -Teorik düşünme süreçlerinin yürütülmesi	1	%30
Final: • İçerik: Dersin tüm içeriğini kapsayan kapsamlı sorular • Format: Yüz yüze. Sınav (60 dakika) • Detaylı Değerlendirme Kriterleri: -Derste işlenen tüm konuların derinlemesine kavranmış olduğunun gösterilmesi -İleri düzey problem çözme becerilerinin kullanılabilmesi	1	%40
Dönem İçi Çalışmaların Başarı Notuna Katkısı		%60
Final Sınavının Başarı Notuna Katkısı		%40
TOPLAM		%100

HAFTALIK KONULAR VE İLGİLİ ÖN HAZIRLIK ÇALIŞMALARI

HAFTALAR	KONULAR	Ön Hazırlık
1	Konu Anlatımı: Programlamaya ve Python'a giriş Sınıf-içi Uygulama (20 dk.): Basit bir “Merhaba Dünya” uygulaması ve temel aritmetik işlemler Sınıf-içi Tartışma (15 dk.): “Neden Python öğreniyoruz?”, “Günlük hayatta hangi alanlarda kullanabiliriz?” soruları üzerine tartışma	1. Python için Anaconda kurulumu. Kaynak: [1]. 2. Python programlama hakkında ön bilgi ve program çalıştırma. Kaynak: Ders Kitabı [1], 1-11. Ders Kitabı [2], 1-10.
2	Konu Anlatımı: Python temelleri (syntax, değişkenler, girdi-çıkı) Sınıf-içi Uygulama (30 dk.): Kullanıcıdan alınan veriler ile bazı basit hesaplamalar yapan programı yazma Sınıf-içi Tartışma (10 dk.): “Farklı veri tiplerini neden tanımlamak zorundayız?” sorusu üzerine tartışma	1. Temel Python değişkenleri, program içerisindeki kullanım durumlarını içeren bölümlerin okunması. Kaynak: Ders Kitabı [1], Bölüm 3, 23-38. Ders Kitabı [4], 7-17.
3	Konu Anlatımı: Akış kontrolü- karar verme mekanizmaları (operatörler, koşullu ifadeler), döngüler Sınıf-içi Uygulama (30 dk.): Karar yapıları ve döngülerle günlük hayata dair küçük senaryoları çözme Sınıf-içi Tartışma (10 dk.): “Koşullu ifadeler ve döngüler günlük hayattaki hangi problemlerin çözümünde kullanılabilir?” sorusu üzerine tartışma	1. Python’daki operatörler, koşullu ifadeler, mantık ifadelerini içeren bölümlerin okunması. Kaynak: Ders Kitabı [1], 39-86. Ders Kitabı [2], 65-73.
4	Konu Anlatımı: İleri seviye veri yapıları (list, tuple veri, dictionary veri)	1. Veri yapıları: Liste, demet ve sözlük yapılarının incelenmesi. Kaynak: Ders



	Sınıf-içi Uygulama (15 dk): Listeler ve sözlükler kullanarak küçük veri setleri üzerinde işlem yapma Sınıf-içi Tartışma (10 dk.): “Hangi durumda liste, hangi durumda sözlük kullanmalıyız?” sorusu üzerine tartışma Kısa Sınav 1 (25 dk.): Ders sonunda, derste işlenen konuları içeren bir kısa sınavın yapılması	Kitabı [2], 363-378; 389-398.
5	Konu Anlatımı: Modüler Python (fonksiyonlar), değişkenlerin kapsamı ve matematiksel işlemler Sınıf-içi Uygulama (20 dk.): Belirli problemleri çözmek için fonksiyonlar tasarlama Sınıf-içi Tartışma (10 dk.): “Fonksiyon kullanmak kodumuzu nasıl daha düzenli hale getirir?” sorusu üzerine tartışma	1. Python’da fonksiyonları içeren bölümlerin okunması. Kaynak: Ders Kitabı [2], 117-148.
6	Konu Anlatımı: Nesneye yönelik programlama Sınıf-içi Uygulama (20 dk.): Gerçek hayattan nesneleri sınıflar ile modelleme Sınıf-içi Tartışma (10 dk.): “Gerçek hayattaki nesneleri Python’da nasıl modelliyoruz?” sorusu üzerine tartışma Kısa Sınav 2 (15 dk.): Ders sonunda, derste işlenen konuları içeren bir kısa sınavın yapılması	1. Nesneye Yönelik Programlama sınıf yapısı ve kalıtım konularını içeren bölümlerin okunması. Kaynak: Ders Kitabı [2], 175-231.
7	Konu Anlatımı: Python’da hata ve hata yönetimi Sınıf-içi Uygulama (20 dk.): Hatalı kod parçalarını analiz edip düzeltme çalışmaları Sınıf-içi Tartışma (5 dk.): “Bir programın hatalara dayanıklı olması neden önemlidir?” sorusu üzerine tartışma	1. Hata ve hata yönetimi konularını içeren bölümlerin okunması. Kaynak: Ders Kitabı [2], 263-279.
8	Ara Sınav 1	Sınav haftasına kadar işlenen konuların tümünün tekrar edilmesi
9	Konu Anlatımı: Python ile Neler Yapabilirim? Sınıf-içi Uygulama (20 dk.): Çalışan küçük bir otomasyon yazma Sınıf-içi Tartışma (15 dk.): “Python’ı kendi alanımızda hangi amaçlarla kullanabiliriz?” sorusu üzerine tartışma	1. Python’un en çok kullanıldığı alanlarla ilgili araştırma yapılması.
10	Konu Anlatımı: Dizi tanımlama ve kullanımı- Numpy Sınıf-içi Uygulama (15 dk.): Diziler üzerinde matematiksel işlemler yapma Sınıf-içi Tartışma (5 dk.): “Listeler ile NumPy dizileri arasındaki fark nedir?” sorusu üzerine tartışma Kısa Sınav 3 (20 dk.): Ders sonunda, derste işlenen konuları içeren bir kısa sınavın yapılması	1. Dizi ve vektör tanımlama, Numpy’nin temellerini içeren bölümlerin okunması. Kaynak: Ders Kitabı [3], 83-121.
11	Konu Anlatımı: Veri analizi- Pandas Sınıf-içi Uygulama (20 dk.): CSV dosyasından veri okuyup basit analiz yapma. Sınıf-içi Tartışma (5 dk.): “Ham veriyi tabloya dönüştürmek neden önemlidir?” sorusu üzerine tartışma	1. Veri analizi ve Pandas uygulamalarını içeren bölümlerin okunması. Kaynak: Ders Kitabı [3], 123-173.
12	Konu Anlatımı: Veri analizi- Pandas Sınıf-içi Uygulama (20 dk.): Bir veri seti üzerinde grupta ve istatistiksel özet çıkarma	1. Veri analizi ve Pandas uygulamalarını içeren bölümlerin okunması. Kaynak: Ders Kitabı [3], 123-173.



	Sınıf-içi Tartışma (5 dk.): “Veri temizleme neden en kritik aşamalardan biridir?” sorusu üzerine tartışma	
13	Konu Anlatımı: Veri Görselleştirme- Grafik Çizimi- Matplotlib Sınıf-içi Uygulama: (20 dk.) Basit grafikler çizerek verileri görselleştirme Sınıf-içi Tartışma: (5 dk.) “Verileri grafiklerle sunmanın avantajları nelerdir?” sorusu üzerine tartışma	1. Veriyi grafiğe aktarma, program üzerinde grafik çizimini içeren bölümlerin okunması. Kaynak: Ders Kitabı [3], 282-317.
14	Konu Anlatımı: Kapsamlı uygulamalar Sınıf-içi Uygulama: (45 dk) Mini veri analizi projesi: NumPy + Pandas + Matplotlib bir arada Sınıf-içi Tartışma: (5 dk.) “Gerçek hayatta bir veri analizi süreci hangi aşamalardan oluşur?” sorusu üzerine tartışma	1. İlgilenilen bir alan ile ilgili küçük bir veri seti araştırılması. 2. Derste işlenen konularla birlikte proje geliştirilip incelenmesi ve tekrardan geliştirilebilirliğine bakılması.
15	Öğrencilerin kendi mini projelerini sınıfta sunması Sınıf-içi Uygulama (15 dk.): Kod + çıktı + grafiklerle birlikte açıklama Sınıf-içi Tartışma (20 dk.): Projelerden elde edilen bulguların güçlü ve zayıf yönlerinin tartışılması	1. Önceden verilen ödevin Python kodlarının hazırlanması, örneklendirilmesi ve raporunun hazırlanması.
16	Final	İşlenen konuların tümünün tekrar edilmesi

AKTS İŞYÜKÜ TABLOSU

Etkinlikler	Sayı	Süresi (Saat)	Toplam İşyükü
Ders Saati	14	3	42
Laboratuvar			
Uygulama			
Arazi Çalışması			
Sınıf Dışı Ders Çalışması	14	3	42
Derse Özgü Staj			
Ödev	1	15	15
Küçük Sınavlar/Stüdyo Kritiği	3	2	6
Projeler			
Sunum / Seminer	1	5	5
Ara Sınavlar (Sınav Süresi + Sınav Hazırlık Süresi)	1	20	20
Final (Sınav Süresi + Sınav Hazırlık Süresi)	1	20	20
Toplam İş yükü:			150
Toplam İş yükü / 30(s):			5.00
AKTS Kredisi:			5



COURSE INFORMATION FORM

FACULTY / GRADUATE SCHOOL	Faculty of Arts and Sciences
DEPARTMENT / PROGRAMME	Mathematics
TITLE OF COURSE	Computer Programming 2
CODE	MAT4560
LOCAL CREDIT	3
ECTS	5
LECTURE HOUR / WEEK	3
PRACTICAL HOUR / WEEK	0
LABORATORY HOUR / WEEK	0
PREREQUISITE	None
SEMESTER	Fall, Spring
COURSE LANGUAGE	English, Turkish
LEVEL OF COURSE	First Cycle
COURSE TYPE	Elective @ Bachelor Programme in Mathematics Elective @ Bachelor Programme in Science Education
COURSE CATEGORY	Major Area Courses
MODE OF DELIVERY	Face-to-Face
OWNER ACADEMIC UNIT	Department of Mathematics
COURSE COORDINATOR	Sema AKIN BAŞ
ASSISTANT(S)	
COURSE OBJECTIVES	This course aims to help students progress from the basics to an advanced level in the Python programming language, starting from scratch. The course also aims to equip students with the skills to understand a given problem and prepare its algorithm, to code the prepared algorithm in the Python programming language in the most appropriate way, and to debug the written program for errors in its algorithm and during compilation.
COURSE CONTENT	The syntax rules of the programming language covered (Python); Python commands; data structures in Python, variables, operators, and basic control structures (if, for, while); functions in Python; principles of modular programming; Pandas and NumPy for data analysis in Python; data analysis applications with Python, and Matplotlib for data visualization in Python. Additionally, students understand the algorithms of some problems or methods assigned as homework, prepare and present their assignments by coding them.
RECOMMENDED OR REQUIRED READINGS	Coursebook: [1] Çobanoğlu, Bülent. <i>Herkes için Python</i> . Pusula Yayınevi, 2022. [2] Hunt , John. <i>A Beginners Guide to Python3 Programming</i> . Springer, 2020. [3] McKinney, Wes. <i>Python for Data Analysis</i> . O'Reilly Media, Inc., 2022. [4] Padmanabhan T. R. <i>Programming with Python</i> . Springer, 2016. Required Readings: [1] https://www.anaconda.com/download Recommended Readings:



Hunt, John. *Advanced Guide to Python 3 Programming*, Springer, 2019.
Shaw, Zed A. *Learn Python the Hard Way*. Addison-Wesley Professional, 2012.

Course Learning Outcomes

Upon successful completion of the course, students will be able to

1. Operate the compiler editor.
2. Apply commands in accordance with the rules of the programming language.
3. Determine a solution method for a problem and code it.
4. Debug the prepared program.
5. Perform data analysis and data visualization applications.
6. Perform basic Python programming that lays the groundwork for data science and machine learning.

EVALUATION SYSTEM

Activities	Number	Percentage of Grade
Attendance/Participation		
Laboratory		
Application		
Field Work		
Special Course Internship (Work Placement)		
Quizzes/Studio Critics: • Content: Comprehensive questions covering all topics addressed up to the exam week • Format: Face-to-face. Multiple choice / fill-in-the-blank quiz (15-30 minutes) • Detailed Assessment Criteria: -Ability to solve problems related to the theoretical topics covered in the course	3	%15
Homework Assignments: • Content: Assignment requiring critical analysis of fundamental concepts covered in class and identification of examples of these concepts within both intra-disciplinary and interdisciplinary contexts • Format: Written reports and group presentations • Detailed Assessment Criteria: -Ability to logically and accurately demonstrate the problem-solving process -Ability to find practical examples of the concepts - Ability to carry out processes of applied thinking, interpretation, and justification	1	%5
Presentations/Jury: • Content: Students will be asked to evaluate their own learning processes and deliver group presentations • Format: Group presentations • Detailed Assessment Criteria:	1	%10



-Ability to accurately explain the topics learned -Proper use of presentation techniques		
Project		
Seminar/Workshop		
Midterms: Content: Comprehensive questions covering all topics addressed up to the exam week Format: Face-to-face written exam. (60 minutes). Detailed Assessment Criteria: -Demonstration of understanding of the fundamental concepts of the course -Ability to solve problems related to theoretical topics -Ability to carry out theoretical reasoning processes	1	%30
Final: Content: Comprehensive questions covering the entire content of the course Format: Face-to-face written exam. (60 minutes). Detailed Assessment Criteria: -Ability to apply advanced problem-solving skills -Demonstration of a thorough understanding of all topics covered in the course	1	%40
Percentage of In-Term Studies		%60
Percentage of Final Examination		%40
TOTAL		%100

WEEKLY SUBJECTS AND RELATED PREPARATION STUDIES

WEEKS	COURSE OUTLINE	Related Preparation
1	Lecture: Introduction to programming and Python Practice (20 minutes): A simple "Hello World" application and basic arithmetic operations In-Class Discussion (15 minutes): Discussion on the questions "Why are we learning Python?", "In which areas of daily life can we use it?"	1. Anaconda installation for Python. Source: [1]. 2. Recollection and activation of background information on Python programming and running programs. Sources: Coursebook [1], 1-11. Coursebook [2], 1-10.
2	Lecture: Python fundamentals (syntax, variables, input - output) Practice (30 minutes): Writing a program that performs some simple calculations using data received from the user In-Class Discussion (15 minutes): Discussion on the question "Why do we need to define different data types?"	1. Reading the sections on basic Python variables and their usage within programs. Sources: Coursebook [1], 23-38. Coursebook [4], 7-17.
3	Lecture: Control flow- decision making mechanisms (operators, conditional statements), loops Practice (30 minutes): Solving small scenarios from daily life using decision structures and loops	1. Reading the sections on operators, conditional statements, and logical expressions in Python. Sources: Coursebook [1], 39-86. Coursebook [2], 65-73.



	In-Class Discussion (10 minutes): Discussion on the question “Which problems in daily life can be solved using conditional statements and loops?”	
4	Lecture: Advanced data structures (list, tuple, dictionary) Practice (15 minutes): Performing operations on small data sets using lists and dictionaries In-Class Discussion (15 minutes): Discussion on the question “When should we use a list, and when should we use a dictionary?” Quiz 1 (25 minutes): A quiz at the end of the class covering the topics taught during the lesson	1. Data structures: Examining list, tuple, and dictionary structures. Source: Coursebook [2], 363-378; 389-398.
5	Lecture: Modular Python (functions), scope of variables and mathematical operations Practice (20 minutes): Designing functions to solve specific problems In-Class Discussion (10 minutes): Discussion on the question “How does using functions make our code more organized?”	1. Reading the sections on functions in Python. Source: Coursebook [2], 117-148.
6	Lecture: Object oriented programming Practice (20 minutes): Modeling real-life objects with classes. In-Class Discussion (10 minutes): Discussion on the question “How do we model real-life objects in Python?” Quiz 2 (15 minutes): A quiz at the end of the class covering the topics taught during the lesson	1. Reading the sections on object-oriented Programming class structure and inheritance. Source: Coursebook [2], 175-231.
7	Lecture: Exception & Error Handling in Python Practice (20 minutes): Analyzing and correcting faulty code snippets. In-Class Discussion (5 minutes): Discussion on the question “Why is it important for a program to be resilient to errors?”	1. Reading the sections covering errors and error management. Source: Coursebook [2], 263-279.
8	Midterm 1	Review of all topics covered up to the exam week.
9	Lecture: What Can I Do with Python? Practice (20 minutes): Writing a small working automation In-Class Discussion (15 minutes): Discussion on the question “How can we use Python in our field?”	1. Research on the most common areas of Python usage.
10	Lecture: Array Definition and Usage - Numpy Practice (15 minutes): Performing mathematical operations on arrays In-Class Discussion (5 minutes): Discussion on the question “What is the difference between lists and NumPy arrays?” Quiz 3 (20 minutes): A short quiz at the beginning of the lesson, covering topics assigned as pre-class preparation within the flipped learning framework	1. Reading the sections on array and vector definition, which cover the fundamentals of Numpy. Source: Coursebook [3], 83-121.



11	Lecture: Data Analysis- Pandas Practice (20 minutes): Reading data from a CSV file and performing basic analysis In-Class Discussion (5 minutes): Discussion on the question “Why is it important to convert raw data into a table?”	1. Reading the sections on data analysis and Pandas applications. Source: Coursebook [3], 123-173.
12	Lecture: Data Analysis- Pandas Practice (20 minutes): Grouping and extracting statistical summaries from a dataset In-Class Discussion (5 minutes): Discussion on the question “Why is data cleaning one of the most critical stages?”	1. Reading the sections on data analysis and Pandas applications. Source: Coursebook [3], 123-173.
13	Lecture: Data visualization - graph drawing Practice (20 minutes): Visualizing data by drawing simple graphs In-Class Discussion (5 minutes): Discussion on the question “What are the advantages of presenting data with graphs?”	1. Reading the sections on transferring data to graphs and drawing graphs on the program. Source: Coursebook [3], 282-317.
14	Lecture: Comprehensive applications Practice (45 minutes): Mini data analysis project: NumPy + Pandas + Matplotlib together In-Class Discussion (5 minutes): Discussion on the question “What stages does a real-life data analysis process consist of?”	1. Researching a small data set related to an area of interest. 2. Developing and reviewing a project based on the topics covered in class and examining its redevelopability.
15	Students present their own mini projects in class Practice (15 minutes): Explanation with code + output + graphs In-Class Discussion (20 minutes): Discussion of the strengths and weaknesses of the findings from the projects	1. Preparing Python code for the previously assigned homework, providing examples, and preparing a report.
16	Final	Review of all topics covered.

ECTS WORKLOAD TABLE

Activities	Number	Duration (Hour)	Total Workload
Course Hours	14	3	42
Laboratory			
Application			
Field Work			
Study Hours Out of Class	14	3	42
Special Course Internship (Work Placement)			
Homework Assignments	1	15	15
Quizzes/Studio Critics	3	2	6
Project			



Presentations / Seminar	1	5	5
Mid-Terms (Examination Duration + Examination Prep. Duration)	1	20	20
Final (Examination Duration + Examination Prep. Duration)	1	20	20
Total Workload:			150
Total Workload / 30(h):			5.00
ECTS Credit:			5



Ders Öğrenim Çıktısı & Program Çıktısı Matrisi

	DÖC-1	DÖC-2	DÖC-3	DÖC-4	DÖC-5	DÖC-6
PC-1 Geometri, analiz, lineer cebir, soyut matematik, cebir ve diferansiyel denklemler gibi matematiğin temel alanlarındaki tanımları ve teoremleri ifade edebilecek, bunlar arasında ilişki kurabilecek ve teoremlerin uygulamalarını açıklayabileceklerdir. / State the definitions and theorems in fundamental areas of mathematics such as geometry, analysis, linear algebra, abstract mathematics, and differential equations, establish connections between them and explain their applications.	=	=	=	=	=	=
PC-2 Matematik alanındaki teorik ve uygulamalı bilgilerini ve matematiksel düşünme becerilerini kullanarak standart ve karmaşık matematiksel problemleri tanımlayabilecek ve analitik ve/veya sayısal yöntemlerle çözebilecek ve matematiksel bir önermenin geçerliliğini farklı ispat yöntemleri ile ispatlayabileceklerdir. / Identify standard and complex mathematical problems, solve these problems with analytical and/or numerical methods and prove the validity of a mathematical proposition through various proof techniques using their theoretical and applied knowledge of mathematics and mathematical thinking skills.	=	=	<u>3</u>	=	=	=
PC-3 Matematik alanındaki teorik ve uygulamalı bilgilerini ve matematiksel düşünme becerilerini kullanarak gerçek hayattaki problemlerin matematiksel modellerini geliştirebilecek ve bu modelleri kullanarak problemlere geçerli çözümler üretebileceklerdir. / Develop mathematical models for real-life problems and generate valid solutions based on these models using their theoretical and applied knowledge of mathematics and mathematical thinking skills.	=	=	<u>5</u>	<u>3</u>	<u>4</u>	=
PC-4 Disiplinlerarası bir	<u>5</u>	<u>5</u>	<u>5</u>	<u>5</u>	<u>5</u>	<u>5</u>



yaklaşım, farklı alanlarda edinmiş oldukları bilgileri sentezleyebileceklerdir. / Synthesise knowledge acquired from different disciplines through an interdisciplinary approach.						
PC-5 Matematik alanında edindikleri bilgi birikimlerini ve matematiksel düşünme becerilerini, teorik matematik, yazılım, finans ve yönetim matematiği, biyomatematik, veri bilimi ve finansal istatistik yöntemleri gibi disiplin-içi ve disiplinlerarası uzmanlık alanlarında geliştirebileceklerdir. / Advance their acquired knowledge in mathematics and mathematical thinking skills in both disciplinary and interdisciplinary areas of specialisation such as pure mathematics, software, financial and managerial mathematics, biomathematics, data science, and financial statistical methods.	<u>3</u>	<u>3</u>	<u>3</u>	<u>3</u>	<u>3</u>	<u>3</u>
PC-6 Matematik alanında yaygın olarak kullanılan en az bir programlama dili ile bilgisayar ve yapay zekâ teknolojilerini, problemleri çözmek, veri analizi yapmak ve simülasyonlar gerçekleştirmek için kullanabileceklerdir. / Use at least one programming language and computer and artificial intelligence technologies widely employed in mathematics for problem-solving, data analysis, and simulations.	<u>5</u>	<u>5</u>	<u>5</u>	<u>5</u>	<u>5</u>	<u>5</u>
PC-7 Matematik ve ilgili alanlardaki bilimsel ve teknolojik gelişmeleri izleyebilecek, kariyer fırsatlarını değerlendirerek kişisel ve mesleki gelişim hedeflerini belirleyebilecek ve bu hedeflere ulaşmak için hayat boyu öğrenme stratejilerini kullanabileceklerdir. / Follow scientific and technological developments in mathematics and related fields, assess career opportunities, identify personal and professional development goals, and adopt lifelong learning strategies to achieve these goals.	=	=	=	=	=	=
PC-8 Bilimsel	=	=	=	=	=	=



arařtırmalarını ve mesleki faaliyetlerini yrtrken doęabilecek hukuksal sonuları ve toplumsal etkileri dikkate alarak mesleki etik ilkeler, kalite standartları ile evrensel deęerler doęrultusunda ve sosyal sorumluluk bilinci ve adalet duygusuyla hareket edebileceklerdir. / Act with a sense of social responsibility and justice and in accordance with professional ethical principles, quality standards, and universal values by taking into account potential legal and societal consequences of their scientific research and professional activities.						
PC-9 Bireysel olarak ya da takımlarda etkin biimde alıřabileceklerdir. / Work effectively both independently and as part of a team.	<u>5</u>	<u>5</u>	<u>4</u>	<u>3</u>	<u>3</u>	<u>5</u>
PC-10 Matematik alanında gvenilir bilgi kaynaklarına ulařarak literatr taraması yapabilecek ve akademik arařtırma tasarlayıp yrtebileceklerdir. / Access reliable sources of information, conduct literature reviews, and design and carry out academic research in the field of mathematics.	=	=	=	=	=	<u>3</u>
PC-11 Matematiksel konuları, teorileri, ispatları, arařtırmaları ve problem zmlerini, matematik terminolojisi kullanarak tm paydařlara Trke ve İngilizcede szl ve yazılı olarak etkili biimde aktarabileceklerdir. / Effectively communicate mathematical topics, theories, proofs, research, and problem solutions to all relevant stakeholders using appropriate mathematical terminology, both orally and in writing, in Turkish and in English.	=	=	=	=	=	<u>3</u>