



DERS BİLGİ FORMU

FAKÜLTE / ENSTİTÜ ADI	Fen-Edebiyat Fakültesi
BÖLÜM / PROGRAM / ANABİLİM DALI ADI	Fizik Bölümü
DERSİN ADI	Fizikte Sayısal Çözümleme
DERSİN KODU	FIZ4610
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 @ Fizik Lisans Programı
DERSİN KATEGORİSİ	Uzmanlık Alan Dersleri
DERSİN VERİLİŞ ŞEKLİ	Yüz Yüze
DERSİ SUNAN AKADEMİK BİRİM	Fizik Bölümü
DERSİN KOORDİNATÖRÜ	Fedai İNANIR
ASİSTAN(LAR)	
DERSİN AMACI	Fizikte Sayısal Çözümleme dersinin amacı, öğrenciye fizik problemlerinin sayısal yöntemlerle çözülmesi için gerekli teorik ve pratik becerileri kazandırmaktır. Bu ders, öğrencilerin fiziksel sistemlerin matematiksel modellerini oluşturma, bu modelleri sayısal tekniklerle analiz etme ve sonuçları yorumlama yetkinliklerini geliştirmeyi hedefler. Ders kapsamında diferansiyel denklemlerin sayısal çözümleri, lineer cebir uygulamaları, optimizasyon teknikleri ve Monte Carlo simülasyonları gibi konular ele alınır. Ayrıca, öğrencilerin programlama araçlarını kullanarak fizik problemlerine yönelik algoritmalar geliştirmesi ve bu algoritmaları gerçek dünya problemlerine uygulaması sağlanır. Dersin sonunda, öğrenciler karmaşık fiziksel sistemleri modelleme, sayısal hesaplama yöntemlerini uygulama ve elde edilen verileri bilimsel bir yaklaşımla değerlendirme becerisi kazanır.
DERSİN İÇERİĞİ	Sayısal yöntemlere giriş; matematiksel modelleme ve fiziksel sistemler; sayısal kök bulma yöntemleri; lineer denklem sistemlerinin çözümü; diferansiyel denklemlerin sayısal çözümleri; sayısal integrasyon teknikleri; lineer cebir ve matris uygulamaları; optimizasyon problemlerine giriş; kısıtsız optimizasyon yöntemleri; kısıtlı optimizasyon teknikleri; programlama araçlarıyla algoritma geliştirme; sayısal sonuçların analizi ve yorumlanması.
DERS KİTABI / MALZEMESİ / ÖNERİLEN KAYNAKLAR	Ders Kitabı: Karaoğlu, B. <i>Sayısal Fizik</i> . 2. baskı, 2013. Önerilen Kaynaklar: [1] Gezerlis, A. <i>Numerical Methods in Physics with Python</i> . Cambridge University Press, 2020. [2] Burden, R. L., & Faires, J. D. <i>Numerical Analysis</i> . Cengage Learning, 2010. [3] Nocedal, J., & Wright, S. J. <i>Numerical Optimization</i> . Springer, 2006.



- [4] Landau, R. H., Páez, M. J., & Bordeianu, C. C. *Computational Physics: Problem Solving with Python*. Wiley, 2015.
- [5] Süli, E., & Mayers, D. F. *An Introduction to Numerical Analysis*. Cambridge University Press, 2003.

Ders Öğrenim Çıktıları

Bu dersi başarıyla tamamlayan öğrenciler,

1. Sayısal yöntemlerin temel kavramlarını ve fiziksel sistemlerin matematiksel modellenmesini anlayarak, gerçek dünya problemlerini sayısal yaklaşımlarla ifade edebilir.
2. Kök bulma algoritmalarını (örneğin, Newton-Raphson, bisection) uygulayarak, fizik problemlerindeki denklemlerin sayısal çözümlerini hesaplayabilir ve hata analizini yapabilir.
3. Gauss eleme, LU ayrıştırma gibi yöntemlerle lineer denklem sistemlerini çözebilir ve fiziksel sistemlerde (örneğin, devre analizi) bu yöntemleri uygulayabilir.
4. Euler, Runge-Kutta gibi yöntemlerle adi ve kısmi diferansiyel denklemleri sayısal olarak çözebilir ve fizik uygulamalarında (örneğin, hareket denklemleri) simülasyonlar gerçekleştirebilir.
5. Trapezoidal, Simpson kuralları gibi entegrasyon yöntemlerini kullanarak, fiziksel büyüklüklerin (örneğin, iş, momentum) sayısal integralini hesaplayabilir ve hassasiyetini değerlendirebilir.
6. Matris işlemleri, özdeğer ve özvektör hesaplamalarını anlayarak, kuantum mekanik veya titreşim analizleri gibi fizik problemlerinde matris tabanlı yöntemleri uygulayabilir.
7. Optimizasyon kavramlarını ve fiziksel bağlamdaki uygulamalarını (örneğin, enerji minimizasyonu) kavrayarak, temel optimizasyon problemlerini tanımlayabilir.
8. Gradient descent, Newton yöntemi gibi kısıtsız optimizasyon algoritmalarını kullanarak, fizik sistemlerindeki eniyileme problemlerini çözebilir ve sonuçları yorumlayabilir.
9. Lagrange çarpanları, KKT koşulları gibi kısıtlı optimizasyon yöntemlerini uygulayarak, gerçekçi fiziksel kısıtlar altında (örneğin, kaynak sınırlamaları) optimizasyon yapabilir.
10. Python, MATLAB gibi araçlarla sayısal algoritmalar geliştirebilir, kodlayabilir ve fizik problemlerine yönelik yazılımlar oluşturabilir.
11. Sayısal sonuçlardaki hataları (yuvarlama, kesme) analiz ederek, fiziksel yorumlamalar yapabilir ve sonuçların güvenilirliğini değerlendirebilir.
12. Tüm konuları entegre ederek, karmaşık fizik problemlerini sayısal yöntemlerle modelleyip çözebilir, sonuçları görselleştirebilir ve raporlayabilir.

DEĞERLENDİRME SİSTEMİ

Etkinlikler	Sayı	Katkı Payı
Devam/Katılım:		
Laboratuvar:		
Uygulama:		
Arazi Çalışması:		
Derse Özgü Staj:		
Kısa Sınavlar/Stüdyo Kritiği • İçerik: Sınav haftasına kadar işlenen konuların tümünü kapsayan kapsamlı soruların sorulması • Format: Yüz yüze. Verilen problemi çözme ve Python, Matlabda kodunu yazma (5-10 dakika)	5	%30



Detaylı Değerlendirme Kriterleri: -Derste işlenen teorik konular ile ilgili problemleri çözebilme			
Ödev:			
Sunum/Jüri:			
Projeler:			
Seminer/Workshop:			
Ara Sınavlar: İçerik: Sınav haftasına kadar işlenen konuların tümünü kapsayan kapsamlı sorular Format: Yüz yüze. Sınav (90 dakika) Detaylı Değerlendirme Kriterleri: - Dersin temel kavramlarının anlaşıldığının gösterilmesi - Sayısal Fizikle ilgili konularla ilgili problemlerin çözülebilmesi - Algoritmik düşünme süreçlerinin yürütülmesi		1	%30
Final: İçerik: Dersin tüm içeriğini kapsayan kapsamlı sorular Format: Yüz yüze. Sınav (90 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: Sayısal Yöntemlere Giriş.	1. Sayısal yöntemlerin temel kavramlarını içeren bir giriş makalesi okunması (örn., Burden & Faires, Numerical Analysis, Bölüm 1). Hata türleri (yuvarlama, kesme) üzerine kısa bir özet hazırlama.	
2	Konu Anlatımı: Matematiksel Modelleme ve Fiziksel Sistemler.	1. Fiziksel sistemlerin matematiksel modellemesine örnekler incelenmesi (örn., basit harmonik hareket). Bir fizik problemini matematiksel olarak ifade eden bir model taslağı oluşturma.	
3	Konu Anlatımı: Sayısal Kök Bulma Yöntemleri (Bisection, Newton-Raphson).	1. Kök bulma algoritmalarının pseudo-kodlarını okunması. Python'da basit bir kök bulma fonksiyonu yazmayı deneyin (örn., $f(x) = x^2 - 4$). Kısa Sınav 1: Sayısal yöntemlere giriş, hata analizi, bisection yöntemi.	
4	Konu Anlatımı: Sayısal Kök Bulma Yöntemleri (Devam).	1. Secant ve sabit nokta yöntemleri üzerine örnek problemler çözme. Algoritmaların yakınsama hızlarını karşılaştıran bir tablo	



		hazırlama.
5	Konu Anlatımı: Lineer Denklem Sistemlerinin Çözümü (Gauss Eliminasyon).	1. Gauss eliminiyorsun yönteminin adımlarını öğrenme. 3x3 bir lineer denklem sistemini elle çözme. Python/MATLAB'da Gauss eleme algoritmasını inceleme ve yazma.
6	Konu Anlatımı: Lineer Denklem Sistemlerinin Çözümü (LU Ayrıştırma).	1. LU ayrıştırma yöntemini okuma (örn., Numerical Recipes, Bölüm 2). Bir 3x3 matris için LU ayrıştırmasını elle yapmayı;EAGlamayı deneme. Kısa Sınav 2: Kök bulma yöntemleri, lineer denklem sistemleri (Gauss elemiminasyon).
7	Konu Anlatımı: Diferansiyel Denklemlerin Sayısal Çözümleri (Euler Yöntemi).	1. Euler yönteminin teorisini okma. Basit bir diferansiyel denklemi (örn., $y' = -ky$) Euler yöntemiyle sayısal olarak çözmek.
8	Ara Sınav 1	Sınav haftasına kadar işlenen konuların tümünün tekrar edilmesi.
9	Konu Anlatımı: Diferansiyel Denklemlerin Sayısal Çözümleri (Runge-Kutta).	1. Runge-Kutta yönteminin 2. ve 4. dereceden algoritmalarını inceleme. Bir fizik problemini (örn., kütle-yay sistemi) Runge-Kutta ile simüle edilmesi.
10	Konu Anlatımı: Sayısal İntegrasyon Teknikleri.	1. Trapezoidal ve Simpson kurallarını öğrenme. Bir integral problemini (örn., $\int_0^1 x^2 dx$) bu yöntemlerle hesaplanması. Kısa Sınav 3: Euler ve Runge-Kutta yöntemleri, sayısal integrasyon.
11	Konu Anlatımı: Lineer Cebir ve Matris Uygulamaları.	1. Matris işlemlerini (çarpma, ters alma) ve özdeğer hesaplamalarını gözden geçirilmesi. Bir 2x2 matrisin özdeğerlerini elle hesaplanması.
12	Konu Anlatımı: Optimizasyon Problemlerine Giriş. Kısıtsız Optimizasyon Yöntemleri.	1. Optimizasyonun temel kavramlarını okunması (örn., Numerical Optimization by Nocedal). Bir fiziksel optimizasyon problemi örneği tanımlanması. 2. Gradient descent ve Newton yöntemlerini incelenmesi. Python'da basit bir gradient descent algoritması yazılması. Kısa Sınav 4: Matris uygulamaları, optimizasyon kavramları.
13	Konu Anlatımı: Kısıtlı Optimizasyon Teknikleri.	1. Lagrange çarpanları ve KKT koşulları üzerine bir makale okunması. Basit bir kısıtlı optimizasyon problemi çözülmesi.
14	Konu Anlatımı: Programlama Araçlarıyla Algoritma Geliştirme.	1. Python veya MATLAB'da bir sayısal algoritma (örn., kök bulma veya entegrasyon) kodlanması. Kodun doğruluğunu test edilmesi.
15	Konu Anlatımı: Sayısal Sonuçların Analizi ve Yorumlanması.	1. Hata analizi ve sonuçların güvenilirliği üzerine bir makale okunması. Bir sayısal çözümün hata analizini raporlanması. Kısa Sınav 5: Kısıtlı optimizasyon, sayısal sonuç analizi.
16	Final	İşlenen konuların tümünün tekrar edilmesi.

**AKTS İŞYÜKÜ TABLOSU**

Etkinlikler	Sayı	Süresi (Saat)	Toplam İşyükü
Ders Saati	14	2	28
Laboratuvar			
Uygulama	14	2	28
Arazi Çalışması			
Sınıf Dışı Ders Çalışması	14	5	70
Derse Özgü Staj			
Ödev			
Küçük Sınavlar/Stüdyo Kritiği	5	5	25
Projeler			
Sunum / Seminer			
Ara Sınavlar (Sınav Süresi + Sınav Hazırlık Süresi)	1	22	22
Final (Sınav Süresi + Sınav Hazırlık Süresi)	1	25	25
Toplam İşyükü :			196
Toplam İşyükü / 30(s) :			6.6
AKTS Kredisi :			5



COURSE INFORMATION FORM

FACULTY / GRADUATE SCHOOL	Faculty of Arts and Sciences
DEPARTMENT / PROGRAMME	Physics
TITLE OF COURSE	Computational Physics
CODE	FIZ4610
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 Physics
COURSE CATEGORY	Major Area Courses
MODE OF DELIVERY	Face-to-Face
OWNER ACADEMIC UNIT	Department of Physics
COURSE COORDINATOR	Fedai İNANIR
ASSISTANT(S)	
COURSE OBJECTIVES	This course aims to provide students with the theoretical and practical skills necessary for solving physics problems by numerical methods. It aims to develop students' ability to construct mathematical models of physical systems, analyze these models with numerical techniques, and interpret the results. The course covers topics such as numerical solutions of differential equations, applications of linear algebra, optimization techniques, and Monte Carlo simulations. It also enables students to develop algorithms for physics problems using programming tools and apply these algorithms to real-world problems. At the end of the course, students will be able to model complex physical systems, apply numerical computing methods, and evaluate the obtained data with a scientific approach.
COURSE CONTENT	Introduction to Numerical Methods; Mathematical Modeling and Physical Systems; Numerical Root-Finding Methods; Solution of Systems of Linear Equations; Numerical Solutions of Differential Equations; Numerical Integration Techniques; Linear Algebra and Matrix Applications; Introduction to Optimization Problems; Unconstrained Optimization Methods; Constrained Optimization Techniques; Algorithm Development with Programming Tools; Analysis and Interpretation of Numerical Results.
RECOMMENDED OR REQUIRED READING	Coursebook: Karaoğlu, B. <i>Numerical Physics</i> . 2nd ed., 2013. Recommended Readings: [1] Gezerlis, A. <i>Numerical Methods in Physics with Python</i> . Cambridge University Press, 2020. [2] Burden, R. L., & Faires, J. D. <i>Numerical Analysis</i> . Cengage Learning, 2010. [3] Nocedal, J., & Wright, S. J. <i>Numerical Optimization</i> . Springer, 2006.



[4] Landau, R. H., Páez, M. J., & Bordeianu, C. C. *Computational Physics: Problem Solving with Python*. Wiley, 2015.

[5] Süli, E., & Mayers, D. F. *An Introduction to Numerical Analysis*. Cambridge University Press, 2003.

Course Learning Outcomes

- Upon successful completion of the course, students will be able to
1. By understanding the basic concepts of numerical methods and mathematical modeling of physical systems, express real-world problems in numerical approaches.
 2. Compute numerical solutions of equations in physics problems by applying root-finding algorithms (e.g., Newton-Raphson, bisection) and perform error analysis.
 3. Solve systems of linear equations using methods such as Gaussian elimination, LU decomposition and apply these methods to physical systems (e.g., circuit analysis).
 4. Numerically solve ordinary and partial differential equations using Euler, Runge-Kutta methods and perform simulations in physics applications (e.g., equations of motion).
 5. Calculate numerical integration of physical quantities (e.g. work, momentum) using integration methods such as trapezoidal, Simpson's rules and evaluate their accuracy.
 6. Apply matrix-based methods to physics problems such as quantum mechanics or vibration analysis by understanding matrix operations, eigenvalue and eigenvector calculations.
 7. Define basic optimization problems by understanding optimization concepts and their applications in a physical context (e.g., energy minimization).
 8. Optimization problems in physical systems using unconstrained optimization algorithms such as gradient descent, Newton's method and interpret the results.
 9. Optimize under realistic physical constraints (e.g., resource limitations) by applying constrained optimization methods such as Lagrange multipliers, KKT conditions.
 10. Develop and code numerical algorithms using tools such as Python, MATLAB and create software for physics problems.
 11. Analyze errors in numerical results (rounding, truncation), make physical interpretations and evaluate the reliability of the results.
 12. Student will be able to model and solve complex physics problems with numerical methods by integrating all topics, visualize and report the results.

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 quiz (5-10 minutes)	5	%30



Detailed Assessment Criteria: -Ability to solve problems related to the theoretical topics covered in the course		
Homework Assignments		
Presentations/Jury		
Project		
Seminar/Workshop		
Midterms: Content: Comprehensive questions covering all topics addressed up to the exam week Format: Face-to-face written exam. (90 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. (90 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 Numerical Methods.	1. Reading an introductory article on the basic concepts of numerical methods (e.g., Burden & Faires, Numerical Analysis, Chapter 1). Preparation of a short summary on error types (rounding, truncation).
2	Lecture: Mathematical Modeling and Physical Systems.	1. Review examples of mathematical modeling of physical systems (e.g., simple harmonic motion). Sketch a model that expresses a physics problem mathematically.
3	Lecture: Numerical Root Finding Methods (Bisection, Newton-Raphson).	1. Reading pseudo-codes of root-finding algorithms. Try writing a simple root-finding function in Python (e.g., $f(x) = x^2 - 4$). Quiz 1: Introduction to numerical methods, error analysis, bisection method.
4	Lecture: Numerical Root Finding (Continued).	1. Solving example problems on Secant and fixed point methods. Preparing a table comparing the convergence speed of algorithms.
5	Lecture: Solution of Systems of Linear Equations (Gaussian Elimination).	1. Learning the steps of Gaussian elimination method. Solving a 3x3 system of linear equations by hand. Analyzing and writing a Gaussian elimination algorithm in Python/MATLAB.
6	Lecture: Solution of Systems of Linear Equations (LU Decomposition).	1. Read about the LU decomposition method (e.g., Numerical Recipes, Chapter 2). Trying to do LU decomposition for a 3x3 matrix by hand. Short Exam 2: Root-finding methods, systems of linear equations



		(Gaussian elimination).
7	Lecture: Numerical Solutions of Differential Equations (Euler Method).	1. Reading the theory of Euler's method. Solving a simple differential equation (e.g., $y' = -ky$) numerically using Euler's method.
8	Midterm Exam 1	1. Review of all the topics covered until the exam week.
9	Lecture: Numerical Solutions of Differential Equations (Runge-Kutta).	1. Investigation of 2nd and 4th order algorithms of the Runge-Kutta method. Simulating a physics problem (e.g., mass-spring system) with Runge-Kutta.
10	Lecture: Numerical Integration Techniques.	1. Learning trapezoidal and Simpson's rules. Calculating an integral problem (e.g., $\int_0^1 x^2 dx$) with these methods. Quiz 3: Euler and Runge-Kutta methods, numerical integration.
11	Lecture: Linear Algebra and Matrix Applications.	1. Review of matrix operations (multiplication, inversion) and eigenvalue calculations. Calculating the eigenvalues of a 2x2 matrix by hand.
12	Lecture: Introduction to Optimization Problems, Unconstrained Optimization Methods.	1. Reading the basic concepts of optimization (e.g., Numerical Optimization by Nocedal). Defining a physical optimization problem example.
13	Lecture: Constrained Optimization Techniques.	1. Investigate gradient descent and Newton methods. Writing a simple gradient descent algorithm in Python. Quiz 4: Matrix applications, optimization concepts.
14	Lecture: Algorithm Development with Programming Tools.	1. Reading a paper on Lagrange multipliers and KKT conditions. Solving a simple constrained optimization problem.
15	Lecture: Analysis and Interpretation of Numerical Results.	1. Coding a numerical algorithm (e.g., root-finding or integration) in Python or MATLAB. Testing the correctness of the code.
16	Final	Reviewing the topics covered.

ECTS WORKLOAD TABLE

Activities	Number	Duration (Hour)	Total Workload
Course Hours	14	2	28
Laboratory			
Application	14	2	28
Field Work			
Study Hours Out of Class	14	5	70
Special Course Internship (Work Placement)			
Homework Assignments			
Quizzes/Studio Critics	5	5	25
Project			
Presentations / Seminar			
Mid-Terms (Examination Duration + Examination Prep. Duration)	1	22	22
Final (Examination Duration + Examination Prep. Duration)	1	25	25
Total Workload :			196
Total Workload / 30(h) :			6.6
ECTS Credit :			5



Ders Öğrenim Çıktısı & Program Çıktısı Matrisi

	DÖC-1	DÖC-2	DÖC-3	DÖC-4	DÖC-5	DÖC-6	DÖC-7	DÖC-8
PC-1 Sayısal yöntemlerin temel kavramlarını anlama ve fiziksel sistemleri modelleme. Temel bilimler ve fizik alanındaki kuramsal bilgilerini akademik düzeyde analiz, yorum ve problem çözümünde kullanabileceklerdir./ Understand the basic concepts of numerical methods and model physical systems. Use their theoretical knowledge in basic sciences and physics in academic analysis, interpretation and problem solving.	<u>5</u>	<u>5</u>	<u>4</u>	<u>3</u>	<u>4</u>	<u>3</u>	<u>2</u>	<u>2</u>
PC-2 Kök bulma algoritmalarını uygulama ve hata analizi yapma. Uygulamalı bilgilerini fiziksel süreçlerin modellenmesi, deneysel tasarım ve problem çözümünde etkili ve amaca yönelik biçimde kullanabileceklerdir./ Apply root-finding algorithms and perform error analysis. Use their applied knowledge effectively and purposefully in modeling physical processes, experimental design and problem solving.	<u>5</u>	<u>3</u>	<u>5</u>	<u>2</u>	<u>4</u>	<u>4</u>	<u>2</u>	<u>2</u>
PC-3 Lineer denklem sistemlerini çözme ve fiziksel sistemlere uygulama. Kuramsal ve/veya deneysel bilgilerini karmaşık fizik problemlerinin çözümünde etkili biçimde kullanarak, uygun analiz ve modelleme yöntemlerini seçip uygulayabileceklerdir./ Solve systems of linear equations and apply them to physical systems. Use their theoretical and/or experimental knowledge effectively in solving complex physics problems, select and apply appropriate analysis and modeling methods..	<u>5</u>	<u>3</u>	<u>5</u>	<u>2</u>	<u>4</u>	<u>4</u>	<u>2</u>	<u>2</u>
PC-4 Diferansiyel denklemleri sayısal olarak çözme ve simülasyon yapma. Disiplinlerarası bir yaklaşımla, farklı alanlarda edinmiş oldukları bilgileri sentezleyebileceklerdir./ Solve differential equations numerically and perform simulations. Synthesize the knowledge acquired in different fields with an interdisciplinary	<u>5</u>	<u>4</u>	<u>5</u>	<u>3</u>	<u>5</u>	<u>4</u>	<u>2</u>	<u>2</u>



approach.								
PC-5 Sayısal entegrasyon yöntemlerini uygulama ve hassasiyet değerlendirme. Fizik biliminin temel ve uygulamalı alanlarında bilgi birikimlerini ve sayısal modelleme araçlarını geliştirme becerilerini kullanabileceklerdir. / Apply numerical integration methods and evaluate sensitivity. Use their knowledge in fundamental and applied areas of physics and their ability to develop numerical modeling tools.	<u>5</u>	<u>3</u>	<u>5</u>	<u>2</u>	<u>4</u>	<u>4</u>	<u>2</u>	<u>2</u>
PC-6 Matris tabanlı yöntemleri fizik problemlerinde uygulama. Fizik alanında yaygın olarak kullanılan programlama dillerini ve teknolojileri problem çözme, veri analizi ve simülasyonlar için etkin biçimde kullanabileceklerdir./ Apply matrix-based methods to physics problems. Use widely used programming languages and technologies in physics effectively for problem solving, data analysis and simulations.	<u>5</u>	<u>3</u>	<u>5</u>	<u>2</u>	<u>4</u>	<u>4</u>	<u>2</u>	<u>2</u>
PC-7 Optimizasyon kavramlarını anlama ve fiziksel bağlamda tanımlama. Fizik ve ilgili alanlardaki gelişmeleri izleyebilecek ve hayat boyu öğrenme stratejilerini kullanabileceklerdir/ Understand optimization concepts and define them in a physical context. Follow the developments in physics and related fields and use lifelong learning strategies	<u>4</u>	<u>4</u>	<u>4</u>	<u>3</u>	<u>4</u>	<u>3</u>	<u>3</u>	<u>2</u>
PC-8 Kısıtsız optimizasyon algoritmalarını uygulama ve yorumlama. Bilimsel faaliyetlerini yürütürken etik ilkeler, kalite standartları ve sosyal sorumluluk bilinciyle hareket edebileceklerdir. / Apply and interpret unconstrained optimization algorithms. Will be able to act with the awareness of ethical principles, quality standards and social responsibility while conducting their scientific activities.	<u>4</u>	<u>4</u>	<u>5</u>	<u>3</u>	<u>4</u>	<u>3</u>	<u>3</u>	<u>2</u>
PC-9 Kısıtlı optimizasyon yöntemlerini fiziksel kısıtlar altında uygulama. Bireysel olarak ya da takımlarda etkin biçimde çalışabileceklerdir./ Apply constrained optimization methods under physical	<u>4</u>	<u>4</u>	<u>5</u>	<u>3</u>	<u>4</u>	<u>3</u>	<u>3</u>	<u>2</u>



constraints. Work effectively individually or in teams.								
PC-10 Programlama araçlarıyla algoritma geliştirme ve kodlama. Fizik alanında güvenilir bilgi kaynaklarına ulaşarak literatür taraması yapabilecek ve akademik araştırma tasarlayıp yürütebileceklerdir./ Algorithm development and coding with programming tools. Access reliable sources of information in the field of physics, conduct a literature review and design and conduct academic research.	<u>4</u>	<u>3</u>	<u>5</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>3</u>	<u>3</u>
PC-11 Sayısal sonuçların hata analizini yapma ve yorumlama. Fizik konularını ve problem çözümlerini Türkçe ve İngilizce olarak etkili biçimde aktarabileceklerdir./ Analyze and interpret error analysis of numerical results. Effectively communicate physics topics and problem solutions in Turkish and English. .	<u>4</u>	<u>3</u>	<u>5</u>	<u>3</u>	<u>4</u>	<u>4</u>	<u>3</u>	<u>3</u>
PC-12 Karmaşık fizik problemlerini entegre ederek çözümleme ve raporlama. Laboratuvar çalışmalarında veri toplayarak teknik raporlar hazırlayabilecek ve mevcut raporları yorumlayabileceklerdir. / Analyze and report complex physics problems by integrating them. Prepare technical reports by collecting data in laboratory studies and interpret existing reports.	<u>5</u>	<u>5</u>	<u>5</u>	<u>4</u>	<u>5</u>	<u>5</u>	<u>4</u>	<u>4</u>