

GENEL TANIM / GENERAL DESCRIPTION

| | | |
|---|--|---|
| Ders Adı / Course Name | PROGRAMMING LANGUAGES / PROGRAMMING LANGUAGES | |
| Ders Kodu / Course Code | 501002191998 | |
| Ders Türü / Course Type | | |
| Ders Seviyesi / Course Level | First Cycle / First Cycle | |
| Ders Akts Kredi / ECTS | 7.00 | |
| Haftalık Ders Saati (Kuramsal) / Course Hours For Week (Theoretical) | 3.00 | |
| Haftalık Uygulama Saati / Course Hours For Week (Objected) | 2.00 | |
| Haftalık Laboratuar Saati / Course Hours For Week (Laboratory) | 0.00 | |
| Dersin Verildiği Yıl / Year | 2 | |
| Öğretim Sistemi / Teaching System | Face to Face / Face to Face | |
| Eğitim Dili / Education Language | English / English | |
| Ön Koşulu Olan Ders(ler) / Precondition Courses | Yok | None |
| Amacı / Purpose | Bu dersin amacı, öğrencinin programlama dillerinin temelindeki kavramları öğrenmesi ve bu kavramların farklı programlama dillerindeki uygulamalarını değerlendirerek, programlama dilleri alanında kapsamlı bir bakış açısına sahip olmasıdır. | Acquisition of the basic concepts of the programming languages and building a general perspective of the programming languages by inspecting various implementations of these concepts on different languages. |
| İçeriği / Content | Programlama paradigmaları ve tarihçe, Dil Tanımı ve Çevrimi, Bağlama ve Kapsam, Veri Tipleri, Altprogramlar, Program Ayrıştırma, Nesneye Yönelik Programlama, İstisna İşleme,Fonksiyonel Programlama, Mantıksal Programlama. | Programming paradigms and history, Language description and translation, Binding and scope, Data types, Subprograms, Program decomposition, Object oriented programming, Exception handling, Functional programming, Logical programming. |
| Önerilen Diğer Hususlar / Recommended Other Considerations | Yok | None |
| Staj Durumu / Internship Status | Yok | None |
| Kitabı / Malzemesi / Önerilen Kaynaklar / Books / Materials / Recommended Reading | Sebesta, R., "Concepts of Programming Languages", Addison-Wesley Publishing, 2006. Watt, D., "Programming Language Design Concepts", John Wiley & Sons, 2004. | Sebesta, R., "Concepts of Programming Languages", Addison-Wesley Publishing, 2006. Watt, D., "Programming Language Design Concepts", John Wiley & Sons, 2004. |
| Öğretim Üyesi (Üyeleri) / Faculty Member (Members) | Dr. Öğr. Üyesi Ahmet EGESÖY | |

ÖĞRENME ÇIKTILARI / LEARNING OUTCOMES

| | | |
|---|---|---|
| 1 | Programlama dilleri kavramlarını özümsemek. | Having a full grasp of programming language concepts. |
| 2 | Programlama dilleri gerçekleştirme teknolojilerini tanımak. | Having a knowledge of programming language implementation technologies. |
| 3 | Var olan programlama dillerini , tarihçelerini, güçlü ve zayıf yanlarını tanımak. | A knowledge of the existing programming languages together with their history and weak and strong aspects.. |
| 4 | Belirli bir yazılım geliştirme projesi için en uygun programlama dilini seçebilmek. | Ability to choose the most appropriate language for a specific software development project. |
| 5 | Yeni bir programlama dili geliştirmek için gereken temel bilgi ve yetilere sahip olmak. | Having the necessary basic knowledge and ability for developing a new programming language. |
| 6 | Var olan bir programlama dilini gerçekleştirebilmek için gereken temel bilgi ve yetilere sahip olmak. | Having the necessary basic knowledge and ability for implementing an existing programming language. |

HAFTALIK DERS İÇERİĞİ / DETAILED COURSE OUTLINE

| Hafta / Week | | | | | |
|--------------|---|--|-----|--|---------------------------|
| 1 | Teorik Dersler / Theoretical | Uygulama | Lab | Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques | Ön Hazırlık / Preliminary |
| | Giriş: Programlama dilleri paradigmaları ve programlama dillerinin gelişimiyle ilgili tarihçe, programlama paradigmalarna genel bir bakış | FORTRAN, ALGOL, COBOL, vb. dillere ait örnekler | | | |
| | Introduction: Programming languages paradigms and history of programming languages. | Examples from languages including: FORTRAN, ALGOL and COBOL | | | |
| 2 | Teorik Dersler / Theoretical | Uygulama | Lab | Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques | Ön Hazırlık / Preliminary |
| | Dil Tanımı: Programlama dillerinde söz dizim ve anlam tanımlanması. Gramer, BNF, EBNF, Anlam tanımlama. | BNF-EBNF ile ilgili soru çözümü | | | |
| | Language Definition: Syntactical and semantical definitions in programming languages. Grammars, BNF, EBNF, Semantics definition methods | Solutions of BNF and EBNF questions | | | |
| 3 | Teorik Dersler / Theoretical | Uygulama | Lab | Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques | Ön Hazırlık / Preliminary |
| | Dil Tanımı: Programlama dillerinde söz dizim ve anlam tanımlanması. Gramer, BNF, EBNF, Anlam tanımlama. | Operasyonel anlam ve aksiyomatik anlam ile ilgili alıştırmalar | | | |
| | Language Definition: Syntactical and semantical definitions in programming languages. Grammars, BNF, EBNF, Semantics definition methods | Exercises of operational and axiomatic semantics | | | |
| 4 | Teorik Dersler / Theoretical | Uygulama | Lab | Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques | Ön Hazırlık / Preliminary |
| | Dil çevrimi: Programlama dillerinin makine diline çevrim süreci, yorumlayıcılar ve derleyiciler. | Sözcüksel analiz konulu örnek | | | |
| | Language Translation: Translation process of programming languages into machine code, interpreters and compilers | Example about lexical analysis | | | |
| 5 | Teorik Dersler / Theoretical | Uygulama | Lab | Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques | Ön Hazırlık / Preliminary |
| | Temel programlama elemanları: Değişkenler ve özellikleri, işlemciler ve özellikleri, işlemci yükleme. | Değişken tipleri ile ilgili alıştırmalar | | | |
| | Basic Programming Constructs: Variables and their properties, processors and their properties, processor loading. | Exercise about variable types | | | |

| | Teorik Dersler / Theoretical | Uygulama | Lab | Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques | Ön Hazırlık / Preliminary |
|----|--|--|-----|--|---------------------------|
| 6 | | | | | |
| | Bağlama Kavramı: Dinamik ve durağan bağlama. Tip, bellek ve kapsam bağlama. | Tip, bellek ve bağlam kapsama ile ilgili alıştıırma | | | |
| | Binding: Dynamic and static binding. Type, memory and scope binding. | Exercises about type, memory and scope binding | | | |
| 7 | | | | | |
| | Veri Tipleri: Veri tipi kavramı. Basit ve yapısal veri tipleri. Tip denetimi, kuvvetli tipleme, tip dönüşümleri, tip uyumluluğu. | Veri tipleri ile ilgili uygulama geliştirme | | | |
| | Data Types: Data type concept. Basic and structural data types Type checking, strong typing, type conversions, type compatibility. | Application development about data types | | | |
| 8 | | | | | |
| | Veri Tipleri: Veri tipi kavramı. Basit ve yapısal veri tipleri. Tip denetimi, kuvvetli tipleme, tip dönüşümleri, tip uyumluluğu. | Veri tipleri ile ilgili uygulama geliştirme | | | |
| | Data Types: Data type concept. Basic and structural data types Type checking, strong typing, type conversions, type compatibility. | Application development about data types | | | |
| 9 | | | | | |
| | Yapısal Programlama: Atama deyimleri, seçimli deyimler, yinelemeli deyimler | Atama deyimleri, seçimli deyimler, yinelemeli deyimler ile ilgili alıştıırma | | | |
| | Structured Programming: Assignment statements, conditional statements, iterative statements. | Exercises about assignment statements, conditional statements and iterative statements | | | |
| 10 | | | | | |
| | Ara sınav | | | | |
| | Midterm Exam | | | | |
| 11 | | | | | |
| | Altprogramlar: Prosedür ve fonksiyonlar. Parametre aktarımı. Etkinlik kayıtları, özyinelemeli altprogramlar. | Altprogram örnekleri | | | |
| | Subprograms, procedures and functions, parameter passing, activation records and recursive subprograms. | Subprogram examples. | | | |

| | | | | | |
|----|--|--|-----|--|---------------------------|
| 12 | Teorik Dersler / Theoretical | Uygulama | Lab | Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques | Ön Hazırlık / Preliminary |
| | Nesneye Yönelik Programlama: Nesneye yönelik programlama kavramları -soyutlama, bilgi saklama, kalıtım, çokyapılılık, dinamik bağlama. | Nesneye yönelik uygulama geliştirme | | | |
| | Object Oriented Programming: Object Oriented Programming concepts, abstraction, encapsulation, inheritance, polymorphism, dynamic binding. | Object oriented application development | | | |
| 13 | Teorik Dersler / Theoretical | Uygulama | Lab | Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques | Ön Hazırlık / Preliminary |
| | İstisna İşleme: İstisna, istisna oluşturma ve çeşitli dillerde istisna işleme. | İstisna işleme ile ilgili uygulama geliştirme | | | |
| | Exception Handling: Creating and handling exceptions in various languages. | Application development about exception handling. | | | |
| 14 | Teorik Dersler / Theoretical | Uygulama | Lab | Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques | Ön Hazırlık / Preliminary |
| | Fonksiyonel Programlama: Fonksiyonel programlamanın temel kavramları, Lisp ve Scheme. | Fonksiyonel programlama ile ilgili ortam tanıtımı ve örnekler (Scheme) | | | |
| | Functional Programming: Basic concepts of functional programming, Lisp and Scheme | Introduction to functional programming environment and Scheme examples | | | |
| 15 | Teorik Dersler / Theoretical | Uygulama | Lab | Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques | Ön Hazırlık / Preliminary |
| | Mantıksal programlama: Temel kavramlar ve Prolog | Mantıksal programlama ile ilgili örnekler (Prolog) | | | |
| | Logical Programming: Basic concepts and Prolog | Logical programming examples (Prolog) | | | |
| 16 | Teorik Dersler / Theoretical | Uygulama | Lab | Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques | Ön Hazırlık / Preliminary |
| | Final Sınavı | | | | |
| | Final exam | | | | |

DEĞERLENDİRME / EVALUATION

| Yarıyıl (Yıl) İçi Etkinlikleri / Term (or Year) Learning Activities | Sayı / Number | Katkı Yüzdesi / Percentage of Contribution (%) |
|---|---------------|--|
| Ara Sınav / Midterm Examination | 1 | 70 |
| Proje Tasarımı /Yönetimi / Project Design/Management | 1 | 30 |
| Toplam / Total: | 2 | 100 |
| Başarı Notuna Katkı Yüzdesi / Contribution to Success Grade(%): | | 40 |
| Yarıyıl (Yıl) Sonu Etkinlikleri / End Of Term (or Year) Learning Activities | Sayı / Number | Katkı Yüzdesi / Percentage of Contribution (%) |
| Final Sınavı / Final Examination | 1 | 100 |
| Toplam / Total: | 1 | 100 |
| Başarı Notuna Katkı Yüzdesi / Contribution to Success Grade(%): | | 60 |
| Etkinliklerinin Başarı Notuna Katkı Yüzdesi(%) Toplamı / Total Percentage of Contribution (%) to Success Grade: | | 100 |
| Değerlendirme Tipi / Evaluation Type: | | |

İŞ YÜKÜ / WORKLOADS

| Etkinlikler / Workloads | Sayı / Number | Süresi (Saat) / Duration (Hours) | Toplam İş Yüğü (Saat) / Total Work Load (Hour) |
|--|---------------|----------------------------------|--|
| Ara Sınav / Midterm Examination | 1 | 2.00 | 2.00 |
| Final Sınavı / Final Examination | 1 | 2.00 | 2.00 |
| Derse Katılım / Attending Lectures | 14 | 3.00 | 42.00 |
| Problem Çözümü / Problem Solving | 12 | 2.00 | 24.00 |
| Rapor Hazırlama / Report Preparation | 1 | 8.00 | 8.00 |
| Proje Hazırlama / Project Preparation | 1 | 48.00 | 48.00 |
| Ara Sınav İçin Bireysel Çalışma / Individual Study for Mid term Examination | 1 | 42.00 | 42.00 |
| Final Sınavı için Bireysel Çalışma / Individual Study for Final Examination | 1 | 42.00 | 42.00 |
| Toplam / Total: | 32 | 149.00 | 210.00 |
| Dersin AKTS Kredisi = Toplam İş Yüğü (Saat) / 30.00 (Saat/AKTS) = 210.00/30.00 = 7.00 ~ 7.00 / Course ECTS Credit = Total Workload (Hour) / 30.00 (Hour / ECTS) = 210.00 / 30.00 = 7.00 ~ 7.00 | | | |

PROGRAM VE ÖĞRENME ÇIKTISI / PROGRAM LEARNING OUTCOMES

| Öğrenme Çıktıları / Learning Outcomes | Program Çıktıları / Program Outcomes | | | | | | | | | | |
|---|--------------------------------------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|
| | 1.1.1 | 1.1.2 | 1.1.3 | 1.1.4 | 1.1.5 | 1.1.6 | 1.1.7 | 1.1.8 | 1.1.9 | 1.1.10 | 1.1.11 |
| 1.Programlama dilleri kavramlarını özümsemek. / Having a full grasp of programming language concepts. | 4 | | | 3 | | | | | | | |
| 2.Programlama dilleri gerçekleştirme teknolojilerini tanımak. / Having a knowledge of programming language implementation technologies. | 4 | | | 3 | | | | | | | |
| 3.Var olan programlama dillerini , tarihçelerini, güçlü ve zayıf yanlarını tanımak. / A knowledge of the existing programming languages together with their history and weak and strong aspects.. | 5 | | 5 | 5 | | | | | | | |
| 4.Belirli bir yazılım geliştirme projesi için en uygun programlama dilini seçebilmek. / Ability to choose the most appropriate language for a specific software development project. | 5 | | 5 | 5 | | | | | | | |
| 5.Yeni bir programlama dili geliştirmek için gereken temel bilgi ve yetilere sahip olmak. / Having the necessary basic knowledge and ability for developing a new programming language. | 3 | | 5 | | | | | | | | |
| 6.Var olan bir programlama dilini gerçekleştirebilmek için gereken temel bilgi ve yetilere sahip olmak. / Having the necessary basic knowledge and ability for implementing an existing programming language. | 3 | | 5 | | | | | | | | |

Katkı Düzeyi / Contribution Level : 1-Çok Düşük / Very low, 2-Düşük / Low, 3-Orta / Moderate, 4-Yüksek / High, 5-Çok Yüksek / Very high