

0205506 PROGRAMLAMA DİLLERİ

Normal Öğretim

İkinci Öğretim

Güz 2018-2019

**Ders Formatı:** Yüz yüze

**EĞİTMEN BİLGİLERİ**

**Eğitmen:**

**Ünvan:**

**Oda:**

**Telefon:**

**Ofis Saatleri:**

**E-mail:**

**DERS TANIMI**

**Kredisi:** *3 kredi (3+0)*

**AKTS**: *5*

**Zorunlu veya seçmeli:** *Bilgisayar Mühendisliği Öğrencileri için Zorunlu*

**Katalog Açıklaması:** *Bu ders geniş ve derinlemesine programlama dili kavramını tartışma imkânı sağlar. Çeşitli dil yapılarının tasarım konularını sunarak, en yaygın dillerin bazılarının yapıları için tasarım seçeneklerinin ve alternatiflerinin mukayesesi, programlama dilleri kavramının anlaşılabilmesi için gerçek temellerin öğrencilere verilmesi amaçlar.*

**Ön koşul Dersler:** *-*

**Kitap ve gerekli materyaller:** *Kenneth C. Louden and Kenneth A. Lambert “Programming Languages: Principles and Practice, Third Edition”*

*R. Toal, R. Rivera, A. Schneider, and E. Choe, Programming Language Explorations, CRC Press, 2017.*

*Robert W Sebesta, Concepts of Programming Languages, 11th Edition, Addison-Wesley, 2016.*

**Ders Hedefleri**

|  |
| --- |
| ***Bu Dersin Amaçları:***  |
| *1* | *Öğrenciye, birçok dil ve özelliklerinin karşılaştırmalı bir incelemesiyle, programlama dili sözdizimi ve anlambiliminin altında yatan temel kavramların çoğunu öğretmek.* |
| *2* | *Öğrenciye çeşitli yeni programlama dili özelliklerini ve paradigmalarını öğretmek* |
| *3* | *Nesne Tabanlı Programlamada temel nesneleri ve yapıları tanımayı öğretmek* |
| *4* | *Yeni programlama dillerini ve yeni dil özelliklerini analiz etmek ve değerlendirmek* |

**Ders Konuları**

|  |  |
| --- | --- |
| ***No*** | ***Konular*** |
| *1* | *Programlama Dillerine Giriş: Programlama dili nedir? Ne için Kullanılınır? Örneklerle çeşitli programlama dilleri, Tarihçesi ve gelişimi, İyi, kötü ve başarılı diller.* |
| *2* | *Çeşitli Programlama Dillerine Genel Bakış: JavaScript, [Ruby | Python], Java, [Clojure | Haskell | Scala | ML], C and C++, [Go | Rust | Swift], birbirlerine üstünlükleri kullanım alanları.* |
| *3* | *Programalama Dili özellikleri: Dilin matematiksel tanımı; Syntax, semantics ve pragmatics; Syntax özellikleri: CFG, BNF, EBNF, diğer notasyonlar; Anlamsal belirtime bir bakış; Sözdizimi hataları ve statik anlam hataları arasındaki farklar.* |
| *4* | *İsimler ve Bağlamalar: İsminin anlamı ve önemi, bağlanma, kapsam (static ve dynamic, stack, ve heap) kavramları; Uygulama sabitleri,değişkenler,tipler, alt programlar ve modüller; Sığ ve derin bağlanma; Aliasing, aşırı yükleme ve polimorfizm.* |
| *5* | *Türler: static, dinamik, strong, weak,manifest ve implicit yazım; Tip kontrolü, eşdeğerlik, tip baskısı ve tür çıkarımı; İlkel türler, sayılar, metin, numaralandırmalar ve işaretçiler; Tuples (ürünler), unions (toplamlar), diziler, kümeler, akışlar, düzenli ifadeler; Soyut ve genel tipler; Kovaryans, Kontrasivyans ve Invaryans; Kurucu, eşitlik testi ve imha; Bağımlı türler.* |
| *6* | *İfadeler (Akış Kontrolü): Operatör önceliği, bağdaşıklık, aritmetik ve sabitlik; Değerlendirme sırası, kısa devre; Yapılandırılmış ve yapılandırılmamış kontrol akışı; Sıralama, seçim, yineleme, yineleme ve belirsizlik.* |
| *7* | *Altprogramlar: Çalışma zamanı yığını ve aktivasyon kayıtları; Çağrı kuralları; Değişen argümanlar ve geri dönen değerler;* |
| *8* | *Ara Sınav* |
| *9* | *Altprogramlar: Yüksek mertebeden fonksiyonlar ve fonksiyonel programlama; Yeniden kapanışlar; Fonksiyon argümanları için desen eşleştirme; İstisnalar; Eşyordamlar; Genel alt yordamlar.* |
| *10* | *Soyutlama, Kapsülleme ve Nesne Yönelimi: Modüller, soyut veri tipleri; Nesne yöneliminin ilkeleri: kapsülleme, kalıtım ve dinamik yöntem bağlanması;* |
| *11* | *Soyutlama, Kapsülleme ve Nesne Yönelimi: Modüller, soyut veri tipleri; Nesne yöneliminin ilkeleri: kapsülleme, kalıtım ve dinamik yöntem bağlanması;* |
| *12* | *Soyutlama, Kapsülleme ve Nesne Yönelimi: Çoklu kalıtım ile ilgili sorunlar; Uygulama sorunları; Sınıf tabanlı vs prototip tabanlı sistemler; Saf ve karma nesne sistemleri.* |
| *13* | *Eşzamanlılık: Threads vs. events; Asynchronous programlama: callbacks, promises, ve async/await.* |
| *14* | *Eşzamanlılık: Multiprocessing vs. multithreading; İletişim ve senkronizasyon sorunları; Paylaşılan hafıza ve mesaj iletimi; Dil-içsel eşzamanlılık vs. kütüphane tarafından yönetilen eşzamanlılık; Uygulamalar* |

**Dersin Öğrenme çıktıları**

*Bu dersin sonunda öğrenciler;*

* *Farklı teknolojileri karşılaştırarak çeşitli veri iletişim ihtiyaçlarına uygun çözümler sunabilecektir.*
* *Birçok dil ve özelliklerinin karşılaştırmalı bir incelemesiyle, programlama dili sözdizimi ve anlambiliminin altında yatan temel kavramların çoğuna hakim olur*
* *Programlama dilleri oluşturmak için kullanılan genel kavramları tanımlar.*
* *Programlama durumunu etkileyen faktörleri ve komutları karşılaştırır.*
* *Ana programlama dillerinin gelişimini ve temel özelliklerini açıklar.*
* *Yeni programlama dillerini ve yeni dil özelliklerini analiz eder ve değerlendirir.*
* *Nesne Tabanlı Programlamada temel nesneleri ve yapıları tanımlar.*
* *Fonksiyonel programlamada saf fonksiyonel fonksiyonların özelliklerini açıklar.*
* *Mantıksal programlamada kullanılan yapıları ve bileşenleri tanımlar.*

**Değerlendirme Yöntemleri**

|  |  |
| --- | --- |
| *1. Ara Sınav* | 40% |
| *2. Final Sınav* | 60% |

**Profesyonel Bileşenler**

|  |  |
| --- | --- |
| *Mühendislik Konuları* | 100% |
| *Genel Eğitim* | 0% |
| *Matematik ve Temel Bilimleri* | 0% |

**Bu açıklamayı hazırlayan ve hazırlayan kişi ya da kişiler**

*Enes Ayan, Nisan 2018*

**Son Revizyon Tarihi**

*Nisan 2018*