

GENEL TANIM / GENERAL DESCRIPTION

Ders Adı / Course Name	Structural Programming / Structural Programming	
Ders Kodu / Course Code	EYON201	
Ders Türü / Course Type		
Ders Seviyesi / Course Level	Bachelor / Bachelor	
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)	0.00	
Haftalık Laboratuar Saati / Course Hours For Week (Laboratory)	0.00	
Dersin Verildiği Yıl / Year	2	
Öğretim Sistemi / Teaching System	Daytime Class / Daytime Class	
Eğitim Dili / Education Language	Turkish / Turkish	
Ön Koşulu Olan Ders(ler) / Precondition Courses		
Amacı / Purpose	Bu ders yapısal programlama dili kullanarak yazılım geliştirmenin kapsam ve kavramlarının öğrenilmesini sağlamaktadır. Aynı zamanda problem çözümleri için yordamsal programlama dilinin özellik ve yapılarını kullanarak yazılım üretme becerilerinizi geliştirmeye yardımcı olacaktır.	This course provides to learn the scope and concepts of software development using structured programming language. It will also help you develop your software production skills using the features and structures of the procedural programming language for problem solving.
İçeriği / Content	Yapısal programlama paradigması; algoritmik çözüm geliştirme; temel programlama yapıları: sıralı işlem, karar, yinelemeli işlemler; C Programlama Dilinde Veri Tipleri ; Kontrol Deyimleri ; Döngüler ; Diziler ve İşaretçiler ; Çok Boyutlu Diziler ;Karakter Dizileri ; Fonksiyonlar ; Fonksiyon İşaretçileri ; Yerel ve Global Değişkenler ; Yapılar	Structured programming paradigm; algorithmic solution development; basic programming structures: sequential operation, decision, iterative operations; Data Types in C Programming Language; Control Statements ; Loops ; Arrays and Pointers ; Multidimensional Arrays ; Character Arrays ; Functions ; Function Pointers ; Local and Global Variables; Buildings
Önerilen Diğer Hususlar / Recommended Other Considerations		
Staj Durumu / Internship Status		
Kitabı / Malzemesi / Önerilen Kaynaklar / Books / Materials / Recommended Reading	A'dan Z'ye C Kılavuzu - Kaan Aslan - PUSULA YAYINCILIK VE İLETİŞİM C İle Programlama - Harvey M. Deitel, Paul J. Deitel - PALME YAYINCILIK Kaan Aslan, A'da Z'ye C Klavuzu, Pusula Yayıncılık	A'dan Z'ye C Kılavuzu - Kaan Aslan - PUSULA YAYINCILIK VE İLETİŞİM C İle Programlama - Harvey M. Deitel, Paul J. Deitel - PALME YAYINCILIK Kaan Aslan, A'da Z'ye C Klavuzu, Pusula Yayıncılık
Öğretim Üyesi (Üyeleri) / Faculty Member (Members)	Dr. Öğr. Üyesi GÜLSÜM ŞANAL	

ÖĞRENME ÇIKTILARI / LEARNING OUTCOMES

1	Öğrenciler, etkin algoritmalar tasarlayabileceklerdir.	Students will be able to design effective algorithms.
2	Öğrenciler, C programlama dilinin özelliklerini kullanabileceklerdir.	Students will be able to use the features of the C programming language.
3	Öğrenciler, geliştirdikleri bir algoritmayı C dili yardımıyla uygulayabileceklerdir.	Students can apply an algorithm they have developed using the C language.
4	Öğrenciler modüler programlama yeteneklerine sahip olacaklardır.	Students will have modular programming capabilities.
5	Öğrenciler, programlarındaki hataları ayıklayabileceklerdir.	Students will be able to debug their programs.
6	Öğrenciler, okunabilir ve tekrar kullanılabilir kaynak kodlar yazabileceklerdir.	Students will be able to write readable and reusable source codes.

HAFTALIK DERS İÇERİĞİ / DETAILED COURSE OUTLINE

Hafta / Week					
1	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Yazılım nedir? Programlama dillerinin seviyelerine göre sınıflandırılması, Programlama dillerinin değerlemesi, Programlama yaklaşımları, C dilinin özellikleri, C dilinin tarihi				
	What is software? Classification of programming languages according to their level, evaluation of programming languages, programming approaches, properties of the C language, history of the C language				
2	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Çevirici programlar, Derleyiciler, Yorumlayıcılar, Derleyicilere olan gereksinim, Tümlşik geliştirme ortamı, Problemin temsili, Çözüm adımlarının ifadesi				
	Assembler programs, compilers, interpreters, requirement for compilers, integrated development environment, representation of problem, expression of solution steps				
3	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	C programlamaya giriş, C'de veri tipleri, Değişkenler, Sabitler, Sayıların Görüntülenmesi, Yazdırma Alanı genişliği, Veri Girişi				
	Introduction to C Programming, Data Types in C, variables, constants, display of numbers, Print Area width, Data Entry				
4	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Aritmetik Operatörler, Atama Operatörü, Karşılaştırma Operatörleri, Mantıksal Operatörler, Arttırma-eksiltme operatörleri, Tip Dönüşümleri				
	Arithmetic operators, assignment operator, comparison operators, Logical Operators, increment-decrement operators, type castings				
5	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Akış Denetimi; IF Komutu, SWITCH Komutu, Koşul Operatörü				
	Flow control; IF instruction, SWITCH instruction, condition operator				

6	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	For Döngüsü				
	For Loop				
7	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	While Döngüsü, Fonksiyon Tanımı, Fonksiyon Çağırma, Fonksiyonların Prototipi, Başlık Dosyaları, Yerel ve Küresel Değişkenler, Fonksiyonlarda Geçerlilik				
	While Loop, Function Definition, Function Calling, Function Prototype, Header Files, Local and Global Variables, Validation in Functions				
8	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Ara Sınav				
	Midterm				
9	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Dizi Tanımı, Dizilere Değer Atama, Dizilere Veri Girişi, Dizilere İlk Değer Verilmesi, Değişken Sınıfı				
	Array Definition, Assigning Values To Arrays, Entering Data Into Arrays, Assigning initial value to arrays, Variable Class				
10	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Fonksiyonlara Dizi Geçirme, Çok Boyutlu Diziler, İki Boyutlu Dizilere İlk Değer Verilmesi				
	Passing Arrays To Functions, Multidimensional Arrays, Assigning initial value to Two-Dimensional Arrays				
11	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Karakter ve Stringler, Tek Karakter Üzerinde İşlem Yapan Fonksiyonlar, Karakter Dizileri, Stringler, String Giriş/Çıkış Fonksiyonları				
	Character and strings, functions that operate on a single character, character arrays, strings, String input / output functions				

12	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Göstericiler, Adres ve Dolaylı Erişim, Gösterici Kullanımı, Gösterici ve Fonksiyonlar, Dizilerin Fonksiyona Geçirilmesi				
	Pointers, address and indirect access, pointer use, pointers and functions, passing arrays to functions				
13	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Yapı Tanımı, Yapının Elemanlarına Erişim, Yapılara İlk Değer Verilmesi				
	Definition of Struct, Access To The Elements of a struct , assigning initial value to structs				
14	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	İç İç Yapılar, Birleşim				
	Nested Structs, unions				
15	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	100
Toplam / Total:	1	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	2	2.00	4.00
Ara Sınav İçin Bireysel Çalışma / Individual Study for Mid term Examination	2	15.00	30.00
Bireysel Çalışma / Self Study	14	5.00	70.00
Derse Katılım / Attending Lectures	14	3.00	42.00
Final Sınavı / Final Examination	1	2.00	2.00
Final Sınavı için Bireysel Çalışma / Individual Study for Final Examination	1	25.00	25.00
Toplam / Total:	34	52.00	173.00
Dersin AKTS Kredisi = Toplam İş Yüğü (Saat) / 25.00 (Saat/AKTS) = 173.00/25.00 = 6.92 ~ / Course ECTS Credit = Total Workload (Hour) / 25.00 (Hour / ECTS) = 173.00 / 25.00 = 6.92 ~			

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	1.1.12	1.1.13	1.1.14	1.1.15
1.Öğrenciler, etkin algoritmalar tasarlayabileceklerdir. / Students will be able to design effective algorithms.	1	2	2	5	4	4	5	5	3	4	5	4	3	3	4
2.Öğrenciler, C programlama dilinin özelliklerini kullanabileceklerdir. / Students will be able to use the features of the C programming language.	1	2	2	5	4	4	5	5	3	4	5	4	3	3	4
3.Öğrenciler, geliştirdikleri bir algoritmayı C dili yardımıyla uygulayabileceklerdir. / Students can apply an algorithm they have developed using the C language.	1	2	2	5	4	4	5	5	3	4	5	4	3	3	4
4.Öğrenciler modüler programlama yeteneklerine sahip olacaklardır. / Students will have modular programming capabilities.	1	2	2	5	4	4	5	5	3	4	5	4	3	3	4
5.Öğrenciler, programlarındaki hataları ayıklayabileceklerdir. / Students will be able to debug their programs.	1	2	2	5	4	4	5	5	3	4	5	4	3	3	4
6.Öğrenciler, okunabilir ve tekrar kullanılabilir kaynak kodlar yazabileceklerdir. / Students will be able to write readable and reusable source codes.	1	2	2	5	4	4	5	5	3	4	5	4	3	3	4

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