GENEL TANIM / GENERAL DESCRIPTION

Ders Adı / Course Name	Computer Programming / Computer Programming	
Ders Kodu / Course Code	EBLG120	
Ders Türü / Course Type		
Ders Seviyesi / Course Level	Bachelor / Bachelor	
Ders Akts Kredi / ECTS	3.00	
Haftalık Ders Saati (Kuramsal) / Course Hours For Week (Theoretical)	2.00	
Haftalık Uygulama Saati / Course Hours For Week (Objected)	2.00	
Haftalık Laboratuar Saati / Course Hours For Week (Laboratory)	0.00	
Dersin Verildiği Yıl / Year	1	
Öğretim Sistemi / Teaching System	Daytime Class / Daytime Class	
Eğitim Dili / Education Language	Turkish / Turkish	
Ön Koşulu Olan Ders(ler) / Precondition Courses	yok	No
Amacı / Purpose	Dersin amacı öğrencilerin temel programlama yöntemleri ve algoritmik problemler ile tanıştırılması; uygulama ve projeler yordamıyla temel programlama disiplinini kazanmasıdır	The aim of the course is to introduce students to basic programming methods and algorithmic problems; is to gain the basic programming discipline with the application and projects procedure.
İçeriği / Content	Önemli bilgisayar sistemi kavramlarına giriş (Bilgisayar bileşenleri, yazılım, işletim dosya sistemi, programlama dilleri, derleyiciler), Problem çözme (Algoritmalar, sözde kod, akış şeması), Algoritma tasarımı, Basit veri tipleri, Sabitler, Değişkenler, Tanımlayıcılar, İsimlendirilmiş sabitler, Aritmetik operatörler, İlişkisel ve mantıksal operatörler, Şartsal ifadeler, Döngüler (While, Do While), Döngüler (For) ve döngü anahtar sözcükleri, Fonksiyonlar, Özyinelemeli fonksiyonlar, Fonksiyon aşırı yüklemesi, Karakter dizileri, karakter dizi fonksiyonları, Diziler, Çok boyutlu diziler, Dizi sıralama, dizilerde arama, İşaretçiler, Referansla geçirme, değerle geçirme, Fonksiyon işaretçileri, Dosya işlemleri.	Introduction to important computer system concepts (Computer components, software, operating file system, programming languages, compilers), Problem solving (Algorithms, pseudocode, flowchart), Algorithm design, Simple data types, Constants, Variables, Identifiers, Named constants, Arithmetic operators, Relational and logical operators, Conditional expressions, While, Do While, For and loop keywords, Functions, Recursive functions, Function overloading, Strings, string functions, Arrays, Multidimensional arrays, Array sorting, searching in arrays, Pointers, Passing by reference, passing by value, Function pointers, File operations.
Önerilen Diğer Hususlar / Recommended Other Considerations		
Staj Durumu / Internship Status	yok	no
Kitabı / Malzemesi / Önerilen Kaynaklar / Books / Materials / Recommended Reading	C ile Programlama (Yazar: Paul Deitel - Harvey Deitel Çeviri Editörü: Cemil Öz Yayınevi: Palme Yayınevi)	C ile Programlama (Yazar: Paul Deitel - Harvey Deitel Çeviri Editörü: Cemil Öz Yayınevi: Palme Yayınevi)
Öğretim Üyesi (Üyeleri) / Faculty Member (Members)	Dr.Öğr. Aliye Saraç	

ÖĞRENME ÇIKTILARI / LEARNING OUTCOMES

1	C programlama dilinde yer alan yapılar hakkında yeterli bilgi.	Adequate knowledge of structures in the C programming language.
2	Farklı yapıdaki problemleri çözmek için algoritma geliştirebilme becerisi.	Ability to develop algorithms to solve problems of different nature.
3	Teorik ve pratik bilgileri program tasarlama ve gerçekleştirme için kullanabilme becerisi.	Ability to use theoretical and practical knowledge for program design and implementation.
4	Gerçek hayatta ki bir ihtiyaca yönelik masaüstü uygulaması projesini C programlama dili ile tasarlayıp geliştirebilmek, raporlamak ve sunabilmek.	To be able to design, develop, report and present a desktop application project for a real-life need with the C 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
	Programlamanın temel kavramları				
	Basic concepts of programming				
2	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	C'ye giriş, veri türleri, bildirim ve tanımlama, sabitler, Operatörler				
	Introduction to C, data types, declaration and definition, constants, Operators				
3	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Kontrol Yapıları				
	Control Structures				
4	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Kontrol yapıları				
	Control Structures				
5	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Döngüler				
	Loops				

6	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Döngüler				
	Loops				
7	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Diziler				
	Arrays				
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
	Fonksiyonlar				
	Functions				
10	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Fonksiyonlar				
	Functions				
11	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Stringler & İşaretçiler				
	Strings & Pointers				

12	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	İşaretçiler				
	Pointers				
13	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Dosya İşlemleri				
	File Operations				
14	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Proje Sunumları				
	Project Presentations				

DEĞERLENDİRME / EVALUATION

Yarıyıl (Yıl) İçi Etkinlikleri / Term (or Year) Learning Activities	Sayı / Number	Katkı Yüzdesi / Percentage of Contribution (%)
Ara Sınav / Midterm Examination	1	40
Proje Hazırlama / Project Preparation	1	20
Proje Sunma / Project Presentation	1	20
Proje Tasarımı /Yönetimi / Project Design/Management	1	20
Toplam / Total:	4	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 Yükü (Saat) / Total Work Load (Hour)
Ara Sınav / Midterm Examination	1	2.00	2.00
Ara Sınav İçin Bireysel Çalışma / Individual Study for Mid term Examination	1	10.00	10.00
Final Sınavı / Final Examination	1	2.00	2.00
Final Sınavı içiin Bireysel Çalışma / Individual Study for Final Examination	1	10.00	10.00
Ev Ödevi / Homework	3	5.00	15.00
Proje Hazırlama / Project Preparation	1	30.00	30.00
Proje Sunma / Project Presentation	1	1.00	1.00
Proje Tasarımı /Yönetimi / Project Design/Management	1	10.00	10.00
Toplam / Total:	10	70.00	80.00
Dersin AKTS Kredisi = Toplam İş Yükü (Saat) / 25.00 (Saat/AKTS) = 80.00/25.00 = 3.20 ~ / Course ECTS Credit = Total Workload (Hour) / 25.00 (Hour / ECTS) = 80.00 / 25.00 = 3.20 ~			

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.C programlama dilinde yer alan yapılar hakkında yeterli bilgi. / Adequate knowledge of structures in the C programming language.	5	5	5								
2.Farklı yapıdaki problemleri çözmek için algoritma geliştirebilme becerisi. / Ability to develop algorithms to solve problems of different nature.		5	5	5							
3.Teorik ve pratik bilgileri program tasarlama ve gerçekleştirme için kullanabilme becerisi. / Ability to use theoretical and practical knowledge for program design and implementation.	5		5	5							
4.Gerçek hayatta ki bir ihtiyaca yönelik masaüstü uygulaması projesini C programlama dili ile tasarlayıp geliştirebilmek, raporlamak ve sunabilmek. / To be able to design, develop, report and present a desktop application project for a real-life need with the C programming language.	5	5	5	5	5						

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