GENEL TANIM / GENERAL DESCRIPTION

Ders Adı / Course Name	Basics of Programming / Basics of Programming	
Ders Kodu / Course Code	OBLG151	
Ders Türü / Course Type		
Ders Seviyesi / Course Level	Associate / Associate	
Ders Akts Kredi / ECTS	5.00	
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		
Ön Koşulu Olan Ders(ler) / Precondition Courses		
Amacı / Purpose	Bu ders ile öğrencinin, programlama temelleri ile ilgili yeterliklerin kazandırılması amaçlanmaktadır.	This course aims to provide students with competencies related to the basics of programming.
İçeriği / Content	Bu ders kapsamında, programcılığa giriş, kodlama öncesi program akışını tasarlamak, kontrol deyimlerini kullanmak, dizi işlemleri yapmak, alt programlarla çalışmak, dosyalama işlemleri yapmak konuları ele alınacaktır.	In this course, Introduction to programming, designing program flow before coding, using control statements, performing array operations, working with subprograms, filing operations will be discussed.
Önerilen Diğer Hususlar / Recommended Other Considerations		
Staj Durumu / Internship Status		
Kitabı / Malzemesi / Önerilen Kaynaklar / Books / Materials / Recommended Reading	Algoritma Geliştirme ve Programlamaya Giriş- Fahri Vatansever - Pusula Yayıncılık	Algoritma Geliştirme ve Programlamaya Giriş- Fahri Vatansever - Pusula Yayıncılık
Öğretim Üyesi (Üyeleri) / Faculty Member (Members)		

ÖĞRENME ÇIKTILARI / LEARNING OUTCOMES

1	Algoritma ve akış diyagramının düzenli ve verimli çalışması için gerekli şartlara uygun kodlama öncesi program akışını tasarlayabilir.	Be able to design pre-coding program flow according to the requirements required for regular and efficient operation of algorithm and flow diagram.
2	Kod yazma temellerinin şartlarına bağlı kalarak programcılığa giriş işlemlerini yapabilir.	Can perform programming input operations by adhering to the requirements of the code writing basics.
3	Karar ve döngü deyimlerinin düzenli ve verimli çalışması için gerekli şartlara uygun kontrol deyimlerini kullanabilir.	Can use control statements in accordance with the requirements necessary for the regular and efficient operation of decision and loop statements.
4	Dizilerin verimli çalışması için gerekli şartlara uygun dizi işlemlerini yapabilir.	Can perform array operations in accordance with the requirements necessary for the efficient operation of arrays.
5	Değer döndüren ve döndürmeyen alt programların verimli çalışması için gerekli şartlara uygun alt programları kullanabilir.	Can use subprograms that meet the requirements for efficient operation of subprograms that return and do not return values.
6	Sıralı ve rastgele erişimli dosyaların düzenli ve verimli çalışması için gerekli şartlara uygun dosya işlemleri yapabilir.	Can perform file operations in accordance with the requirements necessary for the regular and efficient operation of sequential and random access files.

HAFTALIK DERS İÇERİĞİ / DETAILED COURSE OUTLINE

Hafta / Week					
	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
1	Algoritmaya giriş	Algoritma örnekleri			
	Introduction to algorithm	Algorithm examples			
	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
2	Akış Şeması	Algoritma örnekleri			
	Flow Chart	Algorithm examples			
	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
3	Programlama Araçları, Değişkenler ve Sabitler	Algoritma örnekleri			
	Programming tools, variables and constants	Algorithm examples			
	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
4	Giriş-Çıkış İşlemleri, Operatörler	Algoritma örnekleri			
	Input-Output operations, operators	Algorithm examples			
	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
5	Karar Yapıları	Algoritma örnekleri			
	Decision making	Algorithm examples			

6	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Döngü Kontrolleri	Algoritma örnekleri			
	Loop controls	Algorithm examples			
7	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Döngü kontrolleri	Algoritma örnekleri			
	Loop controls	Algorithm examples			
8	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Tek Boyutlu Diziler	Algoritma örnekleri			
	One-Dimensional Arrays	Algorithm examples			
9	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Çok Boyutlu Diziler	Algoritma örnekleri			
	Multi-Dimensional Arrays	Algorithm examples			
10	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Çok Boyutlu Diziler	Algoritma örnekleri			
	Multi-Dimensional Arrays	Algorithm examples			
11	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Değer Döndürmeyen Alt Programlar	Algoritma örnekleri			
	Subroutines That Do Not Return Values	Algorithm examples			

12	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Değer Döndüren Alt Programlar	Algoritma örnekleri			
	Subroutines That Return Values	Algorithm examples			
13	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Sıralı Dosyalar	Algoritma örnekleri			
	Sequential Files	Algorithm examples			
14	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Rastgele Erişimli Dosyalar	Algoritma örnekleri			
	Random Access Files	Algorithm examples			

DEĞERLENDİRME / EVALUATION

Yarıyıl (Yıl) İçi Etkinlikleri / Term (or Year) Learning Activities	Sayı / Number	Katkı Yüzdesi / Percentage of Contribution (%)
Ara Sınav / Midterm Examination	2	100
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 Yükü (Saat) / Total Work Load (Hour)
Ara Sınav / Midterm Examination	2	1.00	2.00
Ara Sınav İçin Bireysel Çalışma / Individual Study for Mid term Examination	2	10.00	20.00
Bireysel Çalışma / Self Study	14	3.00	42.00
Derse Katılım / Attending Lectures	14	3.00	42.00
Final Sınavı / Final Examination	1	1.00	1.00
Final Sınavı içiin Bireysel Çalışma / Individual Study for Final Examination	1	20.00	20.00
Toplam / Total:	34	38.00	127.00

Dersin AKTS Kredisi = Toplam İş Yükü (Saat) / 25.00 (Saat/AKTS) = 127.00/25.00 = 5.08 ~ / Course ECTS Credit = Total Workload (Hour) / 25.00 (Hour / ECTS) = 127.00 / 25.00 = 5.08 ~

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	1.1.16	1.1.17
1.Algoritma ve akış diyagramının düzenli ve verimli çalışması için gerekli şartlara uygun kodlama öncesi program akışını tasarlayabilir. / Be able to design pre-coding program flow according to the requirements required for regular and efficient operation of algorithm and flow diagram.	3	4	4	3	5	5	4	3	3	5	5	3	1	2	3	2	2
2.Kod yazma temellerinin şartlarına bağlı kalarak programcılığa giriş işlemlerini yapabilir. / Can perform programming input operations by adhering to the requirements of the code writing basics.	3	4	4	3	5	5	4	3	3	5	5	3	1	2	3	2	2
3.Karar ve döngü deyimlerinin düzenli ve verimli çalışması için gerekli şartlara uygun kontrol deyimlerini kullanabilir. / Can use control statements in accordance with the requirements necessary for the regular and efficient operation of decision and loop statements.	3	4	4	3	5	5	4	3	3	5	5	3	1	2	3	2	2
4.Dizilerin verimli çalışması için gerekli şartlara uygun dizi işlemlerini yapabilir. / Can perform array operations in accordance with the requirements necessary for the efficient operation of arrays.	3	4	4	3	5	5	4	3	3	5	5	3	1	2	3	2	2
5.Değer döndüren ve döndürmeyen alt programların verimli çalışması için gerekli şartlara uygun alt programları kullanabilir. / Can use subprograms that meet the requirements for efficient operation of subprograms that return and do not return values.	3	4	4	3	5	5	4	3	3	5	5	3	1	2	3	2	2

6.Sıralı ve rastgele erişimli dosyaların düzenli ve verimli çalışması için gerekli şartlara uygun dosya işlemleri yapabilir. / Can perform file operations in accordance with the requirements necessary for the regular and efficient operation of sequential and random access files.	3	4	4	3	5	5	4	3	3	5	5	3	1	2	3	2	2

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