GENEL TANIM / GENERAL DESCRIPTION

Ders Adı / Course Name	Introduction to Computer Programming / Introduction to Computer Programming	
Ders Kodu / Course Code	ESOF102	
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)	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	1	
Öğretim Sistemi / Teaching System	Daytime Class / Daytime Class	
Eğitim Dili / Education Language	English / Turkish	
Ön Koşulu Olan Ders(ler) / Precondition Courses		
Amacı / Purpose	Bu ders Bilgisayar Programlamaya genel bir giriş sağlayacaktır.	This course will provide a general introduction to Computer Programming.
İçeriği / Content	Donanım, yazılım, bilgisayar programlama, Makine Kodu, Programlama Dilleri, veri türleri, döngüler ve diğer programlama dili soyutlamaları gibi temel konular işlenir. Özellikle, kavramların uygulanmasında dil olarak Dart programlama dili kullanılacaktır. Dersin ilerleyen bölümlerinde, yukarıda belirtilen konular hakkında bilgi sahibi olmak için bazı giriş alıştırma seansları da yapılacaktır.	Essential topics such as hardware, software, computer programming, Machine Code, Programming Languages, data types, loops and other programming language abstractions are covered. In particular, Dart programming language will be used as the language for practicing the concepts. Later in the course, some introductory practice sessions to have insight for aforementioned topics will be covered, as well.
Önerilen Diğer Hususlar / Recommended Other Considerations	Dersler sırasında, genellikle, eğitmenin gözetiminde hem ders verme hem de laboratuar oturumları olacaktır.	During class meetings, generally, there will be both lecturing and lab sessions under instructor's supervision.
Staj Durumu / Internship Status		
Kitabı / Malzemesi / Önerilen Kaynaklar / Books / Materials / Recommended Reading	Lecture Presentations & Notes Introduction to JAVA PROGRAMMING AND DATA STRUCTURES; Y. DANIEL LIANG	Lecture Presentations & Notes Introduction to JAVA PROGRAMMING AND DATA STRUCTURES; Y. DANIEL LIANG
Öğretim Üyesi (Üyeleri) / Faculty Member (Members)		

ÖĞRENME ÇIKTILARI / LEARNING OUTCOMES

1	Öğrenciler, örneğin I/O cihazları, RAM ve CPU gibi bilgisayarlar için bilgisayarların çalışma mekanizmasını açıklayabilir.	Students can explain the working mechanism of computers for computers, for example, the I/O devices, RAM, and CPU.
2	Öğrenciler java'da proje oluşturabilir. Ayrıca değişkenleri ve metotları tanımlayabilirler.	Students can create projects in java. Also, they can define variables and methods.
3	Öğrenciler döngüleri (for, while ve do while) ve koşulları (if ve switch) kullanarak programınızın akışını yönetmek için kod yazabilirler.	Students can write code to manage the flow of your program using loops (for, while, and do while) and conditions (if and switch).
4	Öğrenciler Java'da başlangıç (basit) algoritmalar için kod yazabilirler.	Students can write code for beginner (simple) algorithms in Java.

HAFTALIK DERS İÇERİĞİ / DETAILED COURSE OUTLINE

Hafta / Week					
1	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Bilgisayarlara, Programlara ve Java'ya Giriş				
	Introduction to Computers, Programs, and Java				
2	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Temel Programlama				
	Elementary Programming				
3	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Seçimler 1				
	Selections 1				
4	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Seçimler 2				
	Selections 2				
5	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Matematiksel Fonksiyonlar, Karakterler ve String'ler				
	Mathematical Functions, Characters, and Strings				

6	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	döngüler 1				
	Loops 1				
7	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	döngüler 2				
	Loops 2				
8	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Ara Sınav ve Soru/Cevap				
	Midterm Exam and Question/Answer				
9	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Metodlar				
	Methods				
10	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Programlama Akışları				
	Programming Flows				
11	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Tek Boyutlu Diziler				
	Single-Dimensional Arrays				

12	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Seçimler, Döngüler, Metodlar ve Diziler Kullanarak Program Yürütme Kontrolü				
	Program Execution Control Using Selections, Loops, Methods, and Arrays				
13	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Çok Boyutlu Diziler				
	Multidimensional Arrays				
14	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Nesnelere ve Sınıflara Giriş				
	Introduction to Objects and Classes				

DEĞERLENDİRME / EVALUATION

Yarıyıl (Yıl) İçi Etkinlikleri / Term (or Year) Learning Activities	Sayı / Number	Katkı Yüzdesi / Percentage of Contribution (%)
Ara Sınav İçin Bireysel Çalışma / Individual Study for Mid term 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 Yükü (Saat) / Total Work Load (Hour)
Ara Sınav İçin Bireysel Çalışma / Individual Study for Mid term Examination	5	4.00	20.00
Bireysel Çalışma / Self Study	12	2.00	24.00
Final Sınavı içiin Bireysel Çalışma / Individual Study for Final Examination	5	4.00	20.00
Final Sınavı / Final Examination	1	1.00	1.00
Derse Katılım / Attending Lectures	14	1.00	14.00
Ara Sınav / Midterm Examination	1	1.00	1.00
Toplam / Total:	38	13.00	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.Öğrenciler, örneğin I/O cihazları, RAM ve CPU gibi bilgisayarlar için bilgisayarların çalışma mekanizmasını açıklayabilir. / Students can explain the working mechanism of computers for computers, for example, the I/O devices, RAM, and CPU.	4	2	3	1	2	1	1	1	2	1	1
2.Öğrenciler java'da proje oluşturabilir. Ayrıca değişkenleri ve metotları tanımlayabilirler. / Students can create projects in java. Also, they can define variables and methods.	4	3	3	2	3	1	1	1	3	2	1
3.Öğrenciler döngüleri (for, while ve do while) ve koşulları (if ve switch) kullanarak programınızın akışını yönetmek için kod yazabilirler. / Students can write code to manage the flow of your program using loops (for, while, and do while) and conditions (if and switch).	5	3	4	2	3	1	1	2	4	2	1
4.Öğrenciler Java'da başlangıç (basit) algoritmalar için kod yazabilirler. / Students can write code for beginner (simple) algorithms in Java.	4	3	4	2	3	1	1	1	3	1	1

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